Remote Sensing Applications: Society and Environment 41 (2026) 101818

Contents lists available at ScienceDirect

Remote Sensing Applications: Society and Environment

journal homepage: www.elsevier.com/locate/rsase

Urban tree crown detection based on deep learning and high-resolution aerial imagery: PTCNet for Pullman, WA, USA

Okikiola Michael Alegbeleye *, Arjan Johan Herman Meddens, Yetunde Oladepe Rotimi, Kelechi Godwin Ibeh

School of Environment, Washington State University, Pullman, WA, USA

ARTICLE INFO

Keywords:
Individual tree detection
Urban trees
YOLOv3
Mask R-CNN
LiDAR

ABSTRACT

Individual tree data in urban settings are used for many purposes, and gathering such information requires time and other limited resources. Additionally, the data collected are spatially and temporally sparse, especially for continuous monitoring. However, high-resolution images and deep learning can offer automated and accurate detection of trees in complex urban settings. Therefore, this study compared four popular convolutional neural network CNN-based object detection models (You Only Look Once v3, RetinanNet, Mask R-CNN, and Faster R-CNN) to map individual trees. We used high-resolution aerial imagery (~8 cm spatial resolution), which was manually annotated to derive training (4,859) and testing (1,184) datasets. The analysis was carried out in three phases: First, we trained all the models for 20 epochs and evaluated the performance using standard metrics (Precision, Recall, and F1 score). Second, the best model was selected and retrained longer (30 epochs) with more data (5002 annotations) to develop an urban tree crown detection model for Pullman – a small-sized city in the inland northwest of the United States. Finally, we tested the reliability of the developed model under two scenarios. According to our analysis, YOLOv3 (F1 score: 69 %) outperformed Mask R-CNN (F1 score: 60 %), RetinaNet (F1 score: 57 %), and Faster R-CNN (F1 score: 52 %). Based on the evaluation metrics and visual assessment, YOLOv3 was selected to develop the final urban tree crown detector – Pullman Tree Crown Network (PTCNet), for our study area. PTCNet had precision and recall values of 78 % and 62 %, respectively. It also performed well under different tree arrangements, achieving an F1 score of over 70 %. The model was used to generate ~12,000 individual tree locations. Subsequently, height information was extracted from a LiDAR-derived canopy height model, and a comprehensive tree inventory dataset was derived. The model and dataset are publicly available (https://github.com/Okikiola-Michael/PTCNet) for different applications, thus, contributing to open science. This study provides a straightforward and repeatable framework for researchers and managers to map urban trees with height information, which is useful for spatial and temporal tree monitoring. This study further highlights the performance of four popular models and supports the application of deep learning and aerial imagery for individual tree detection in complex urban settings.

* Corresponding author.
E-mail address: okikiola.alegbeleye@wsu.edu (O.M. Alegbeleye).

https://doi.org/10.1016/j.rsase.2025.101818
Received 15 April 2025; Received in revised form 27 November 2025; Accepted 4 December 2025
Available online 5 December 2025
2352-9385/

1. Introduction

Individual tree data are needed at appropriate spatial and temporal scales for management purposes (Pu, 2021). Such data can provide critical information to urban planners and resource managers to conserve endangered species (Zhao et al., 2023), identify tree conditions (Plowright et al., 2016), collect genetic/botanical data (Wagner et al., 2019), estimate forest parameters such as biomass, timber volume, and carbon (Qian et al., 2023), and monitor biodiversity (Ma et al., 2021). Many public and private organizations (e.g., the US Forest Service) use tree lists to develop models for management purposes (Bechtold and Patterson, 2005). For example, city governments need individual tree locations and attribute data for decision-making on urban green spaces, restoration projects, tree health, and overall land management. However, mapping individual tree species using field surveys is constrained by time, access restrictions, and financial resources (Pu, 2021; Ma et al., 2021; Beloiu et al., 2023). Additionally, repetitive measurements, sampling, and assessments of trees following disturbances may be challenging to achieve using field surveys. The use of remote sensing and deep learning has evolved over the years, providing solutions to these challenges (Gui et al., 2024). However, no robust model or architecture can achieve accurate results across different locations (Zheng and Wu, 2021). This necessitates the need for locally developed models for accurate tree detection.

Over the years, automatic individual tree detection using high-resolution optical images has employed traditional methods such as template matching (Brandtberg, 1999; Huo and Lindberg, 2020), region-growing (Erikson, 2003; Gu et al., 2020), watershed segmentation (Lamar et al., 2005; Liu et al., 2024), valley-following, and multi-scale segmentation (Skurikhin et al., 2013; Wang et al., 2024). These methods were limited in delineating individual trees in terms of the overall accuracy, multi-species mapping, and the transferability of the developed models to another site. Recently, deep learning (convolutional neural networks [CNN] architectures), a branch of machine learning, has been applied to RGB images for image/semantic segmentation (grouping pixels into classes, i.e., pixel-based classification); object detection (geolocating an object and creating a boundary with labels for each object, e.g., tree detection); and instance segmentation (a combination of object detection and image segmentation) (Hoeser and Kuenzer, 2020; Beloiu et al., 2023). CNN architectures have been reported to perform best in analyzing spatial patterns during object detection, especially when very-high spatial resolution data are used compared to other image analysis techniques (Nurul Husna et al., 2019; Braga et al., 2020a; Schiefer et al., 2020; Pu, 2021; Beloiu et al., 2023). This can be attributed to the robust connected neural network architectures (Xi et al., 2020) that enable the algorithm to learn the object-level representation of image features (Weinstein et al., 2019) to detect objects without the need for a preselected feature (Zhao et al., 2023).

Several deep learning algorithms for automatic individual tree detection have been proposed with different architectures. Object detection frameworks based on CNN are categorized into two types: one-stage and two-stage detectors (Weinstein et al., 2019; Zhao et al., 2023). The significant difference lies in the CNN architecture stages for object detection or classification. A one-stage detector performs feature extraction and detection in a single module, while two-stage detectors consist of two modules (a region proposal module and an object/classification module) (Zhao et al., 2023). The region proposal module is a small network that slides over the convoluted feature map to propose object locations with object scores and applies non-maximum suppression to remove redundant boxes. Examples of object detection algorithms include region-based CNNs (RCNN; Girshick et al., 2014), single-shot detectors (SSD; Liu et al., 2016), region-based fully convolutional networks (R-FCN; Dai et al., 2023), You Only Look Once (YOLO; Redmon et al., 2016), and RetinaNet (Lin et al., 2018). They have been used in detecting cash crops (Li et al., 2017; Zheng et al., 2019), live and dead trees (Zhou et al., 2024; Yao et al., 2024), fruit trees (Zhu et al., 2022; Ang et al., 2024), wildlife (Delplanque et al., 2022; Sankaran, 2024) among other applications using multispectral, point cloud, and hyperspectral data (Hoeser et al., 2020).

Many urban tree detection studies have used or modified common CNN models such as YOLO (Itakura and Hosoi, 2020; Yinghua et al., 2021; Choi et al., 2022), Faster RCNN (Xia et al., 2021; ZamboniThgeThe et al., 2021; Zhang et al., 2022), and Mask RCNN (Uematsu et al., 2022; Yang et al., 2022; Zhang et al., 2024), while some studies modified existing models or developed new architectures. Ventura et al. (2024) introduced a point-based tree detection method using a confidence map approach applied to fine-resolution NAIP (National Agricultural Imagery Program) imagery in various California cities. Their developed model outperformed two tree detection methods: PyCrown and Deepforest. Velasquez-Camacho et al. (2023) used two neural networks on different remote sensing images to automatically detect and geo-position urban trees. In addition to vertical information from aerial and satellite images, their study used Google Street View (GSV) images for horizontal information, supporting accurate tree detection. However, the performance of these different algorithms differs because of varying tree structures and conditions (e.g., tree species, crown size, tree health, and understory vegetation), architectures and frameworks, training datasets (image properties, resolution, and lightning), data fusion, and augmentation (Zhao et al., 2023). Hence, there is a need to (1) compare and develop localized tree detection models (live and dead, single and multi-species) useful for end users, (2) develop frameworks to detect small trees and distinguish complex tree crowns, and (3) keep testing the generalization of the CNN architectures (transfer learning) in different urban settings to improve predictability and robustness.

Pullman, Washington, USA, is a small-sized city (population: 32,863) containing a variety of building densities (i.e., structures per unit area) and a diverse urban ecosystem. However, there is no existing automated workflow for detecting tree crowns, nor a database of trees in the city. This has limited understanding of the spatial distribution of trees, tree density, and diversity, and the ecosystem services they provide. However, city planners have collected high-resolution leaf-off aerial imagery from an airplane at 3-year intervals over the last decade. Therefore, there is a clear opportunity to leverage high-resolution aerial imagery, utilize available LiDAR data, and compare deep learning object detection models to develop a robust automated workflow for detecting tree locations within the city. The availability of a custom tree detection model, spatial data on individual trees, and their corresponding heights for the city will enable city planners to make informed management and monitoring decisions.

Therefore, the overall objective was to map individual trees in Pullman using high-resolution aerial imagery and to develop an urban tree inventory with individual tree heights. To achieve this, we compared four object detection models, selected the best model, predicted individual tree locations (using tree crowns), and estimated tree heights. The performance of the developed tree crown detector was further assessed in detecting tree crowns under different tree densities and arrangements. Specific sub-objectives were to.

(1) Determine the most accurate deep learning tree detection algorithm by comparing YOLOv3, RetinaNet, Mask R-CNN, and Faster R-CNN performance in the study area.
(2) Develop a final urban tree crown detector (based on the most accurate algorithm in sub-objective 1) and use it to identify tree locations on a large scale across the city.
(3) Create a tree inventory database with tree location and height using a LiDAR-derived vegetation height model for the city of Pullman.

The uniqueness of our study lies in its application and usefulness to city managers and researchers, providing comprehensive individual tree inventory data to support management decisions and inform future research. While there are existing tree detection models and frameworks for other regions, this study is the first to develop a model that maps individual trees with corresponding inventory data in our study area, highlighting the performance of four popular deep learning models in detecting trees in a complex urban setting.

2. Methods

2.1. Study area

Pullman is a city in Whitman County, located in southeastern Washington, USA, within the Palouse region of the Pacific Northwest (Fig. 1). The city covers approximately 28.8 km^2 with an average elevation of 774 m above sea level. Pullman has a rolling hill topography with four distinct hills, several creeks, and waterways that dissect the city, with Paradise Creek being the most notable creek. Pullman is an ideal case study area, characterized by gentle topography, a relatively high density of trees, the availability of a high-resolution aerial dataset, and a need for a geolocated tree database. The Palouse, where Pullman is located, consists of deep, well-drained soils formed in loess on hills. Pullman contains diverse tree species, including Lodgepole pine (*Pinus contorta),* Douglas fir (*Pseudotsuga menziesii*), Norway maple (*Acer platanoides*), Western redcedar (*Thuja plicata*), Red oak (*Quercus rubra*), and many other ornamental and exotic tree species. The city experiences a temperate climate with a short, warm, and dry summer, a distinct winter period, and mild fall and spring weather. The average annual precipitation is about 518 mm with a mean annual temperature range of −3.3 °C to 28.9 °C.

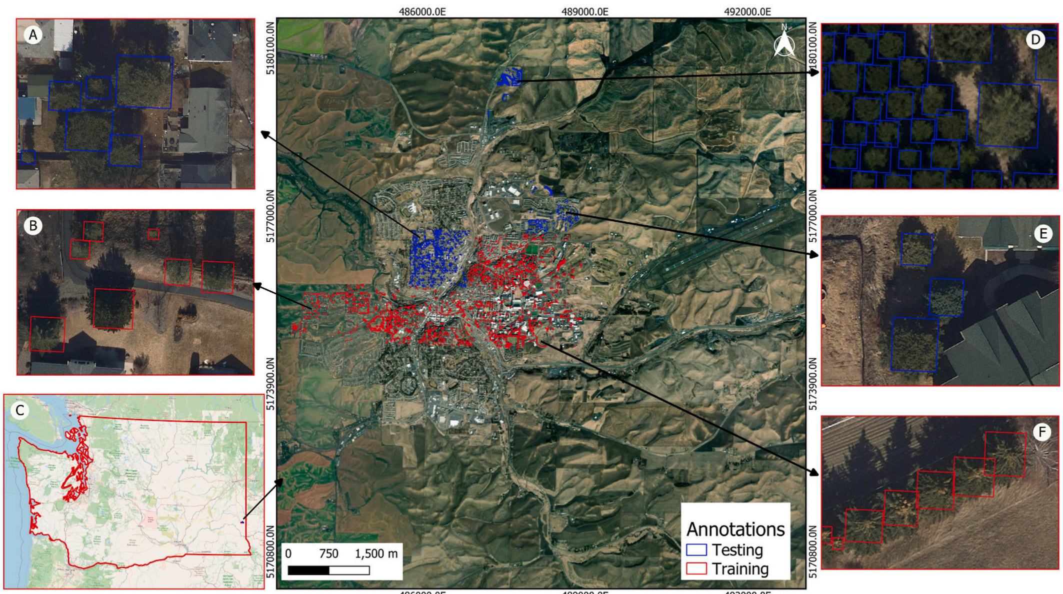

Fig. 1. The city of Pullman (center) with the training (red) and testing hand-annotated tree crowns (blue) data. A: Sample of the test data; B: Training data samples with different tree canopy sizes; C: Map of Washington with Pullman; D: A sample of the clustered tree arrangement used to further test the final developed model; E: Samples of random tree arrangement used to further test the final developed model, and F: Training data samples with closely arranged trees.

2.2. High-resolution aerial imagery

The RGB data was collected using a human-crewed airplane with a large-format Leica Digital Mapping Camera (DMC-3) imaging sensor on April 19, 2023. The vendor processed the images, and the RGB orthomosaic images were divided into 97 (approximately 1 km × 1 km) tiles with ~8 cm spatial resolution (http://bit.ly/3W1SNlX, accessed on 1/22/2025). The city planners primarily collect the RGB images during the leaf-off season to observe ground features. However, we leveraged it to map individual trees. The RGB images had three bands and were projected in the North American Datum (NAD) 1983 StatePlane Washington South FIPS 4602 (US Feet) coordinate reference system. Seven RGB image tiles (1 km x km) with various tree arrangements within the city were purposively selected for training, and one image tile was used for testing the initial tree detection models (Fig. 2). The training data was used to train the model, while the testing data was used as independent data to evaluate the initial model's performance of the four candidate CNN architectures. Additionally, two more locations (or tiles: orthomosaic images) were selected with challenging tree arrangement case studies (1: dense canopy and 2: sparse trees mixed with different urban features) to further assess the model accuracy in detecting tree crowns. The two case studies included areas with (1) random locations of trees near urban development (referred to as RA) and (2) clustered and high tree density (referred to as CA). These case study areas were selected based on the number and arrangement of trees to assess how the algorithms performed under different tree canopy densities. This is important because canopy structure can significantly affect the efficacy of individual tree detection (Wang et al., 2016).

2.3. Annotations

Annotations are vector data (points or polygons) that denote the location and boundaries of the object of interest within an image, which are used to train and test models. All the selected orthomosaic images were carefully hand-annotated by a remote sensing expert with local tree knowledge and the city of Pullman to generate labeled tree crowns (with a square bounding box approximating the tree crown area) as shapefiles in ArcGIS Pro 3.3.1 for all the training and testing image tiles (Table 1). To minimize potential bias, the annotations were reviewed multiple times for accuracy. We annotated all the visible tree crowns and converted them to two deep learning framework annotation formats: PASCAL Visual Object Classes (PASCAL VOC) for Faster R-CNN, YOLOv3, and RetinaNet models, and RCNN Mask for Mask R-CNN.

2.4. Tree detection models

Various deep-learning algorithms with different architectures have been proposed for individual tree detection. To select the best tree-detection model for our study area, we compared the performance of four popular deep learning models. Comparatively, we evaluated two one-stage (YOLOv3 and RetinaNET) and two two-stage (Faster R-CNN and Mask R-CNN) detection models. The models were selected because they were the best object detection models available in the ArcGIS Pro 3.3.1. software that was used for the analysis. The models are described below.

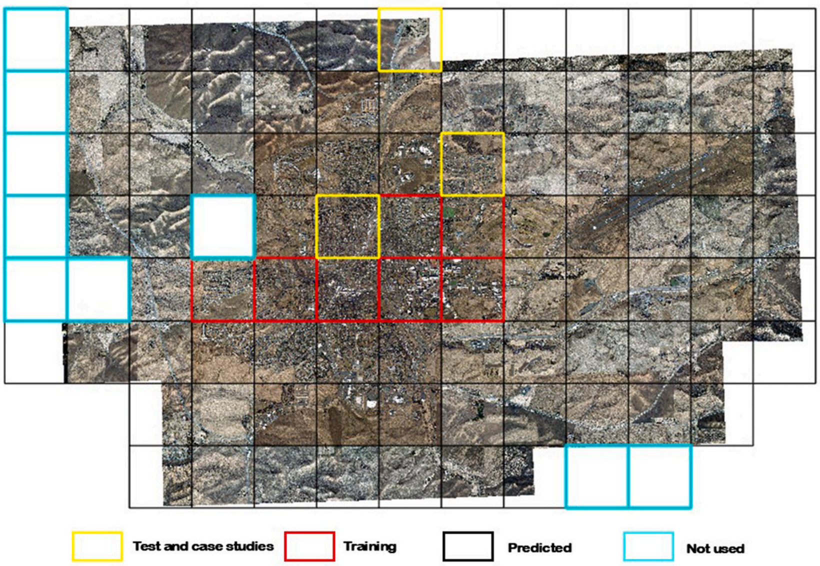

Fig. 2. Image tiles over Pullman indicating the tiles used for training the model, testing the model (including case studies), large-scale prediction, and tiles with no trees.

Table 1
The number of hand-annotated tree crowns used for training and testing the models.

Data description	Number of Annotations
(A) Training/testing data for the initial model	
Preliminary training data for all the initial four models	4859
Testing data for accuracy assessment	1184
(B) Training/testing data for the final model	
New training annotations for the final model	143[a]
Total training data from A used for the final model	(4859[b])
Testing data for the final model accuracy assessment from A	(1184[b])
(C) Testing data for selected case studies	
Random tree arrangements (close to buildings)	404
Clustered tree arrangement (high tree density case)	1194
Total unique annotations used in this study	**7784**

[a] New annotations were from the same area as the initial training data.
[b] Reused from (A) for the final model.

2.4.1. *One-stage detectors*

One-stage detectors are deep learning algorithms that predict bounding boxes and class probabilities in a single pass. We tested two one-stage detectors, YOLOv3 and RetinaNET. YOLOv3 (Redmon et al., 2016) is a fast CNN-based object detector with a Darknet-53 backbone that extracts input image features. It was proposed as a modification to YOLOv2 for better detection accuracy (Pham et al., 2020), and the Darknet-53 backbone was reported to outperform other start-of-the-art backbones, such as ResNet (Redmon and Farhadi, 2018). YOLOv3 detects objects at three scales and has 106 convolutional layers. The convolutional layers are filters or kernels that convolve the input image to extract features or generate feature maps. The YOLOv3 architecture treats object detection as a regression problem by predicting an object score for the bounding box in an image using logistic regression (Redmon and Farhadi, 2018). The neural network divides each image into an $S \times S$ grid, predicts y boxes within each grid, and provides their confidence scores. The non-maximum suppression (NMS) framework is used to select the best-predicted box. YOLOv3 has high detection speed and accuracy (Tian et al., 2019) and has been used and modified to detect different objects in images, such as urban tree species (Li and Yao, 2020; Choi et al., 2022), crops/tree crops (Tian et al., 2019; Hao et al., 2022), and diseased trees (Safonova et al., 2022; Wu et al., 2023). Although newer YOLO variants (e.g., YOLOv7, YOLOv8) offer improved performance, YOLOv3 was selected because it provides a stable, well-validated, and computationally efficient architecture for tree detection.

RetinaNet is also a one-stage detector that was proposed in 2018 (Lin et al., 2018). It is an improvement over other one-stage detector models (e.g., SSD) to demonstrate the loss function created to reduce the extreme foreground-background class imbalance that occurs when training one-stage detectors. RetinaNet consists of a backbone, usually a convolutional network that generates the feature map, and two subnetworks that perform convolutional classification on the feature maps and regression on the bounding boxes (Lin et al., 2018). An example of a tree detection model created using the RetinaNet framework on aerial imagery was the Deepforest package by Weinstein et al. (2019).

2.4.2. *Two-stage detector*

Unlike one-stage detectors, two-stage detectors include an additional region proposal network (RPN) in their CNN architectures to generate regions of interest. Faster R-CNN and Mask R-CNN were the two-stage detectors tested in this study.

Faster R-CNN (Ren et al., 2017) is a modified version of Fast R-CNN (Tanatipuknon et al., 2021) and uses both the RPN and a classification network to detect a region of interest and classify the predicted bounding box, respectively (Bao et al., 2022). Faster R-CNN consists of two modules: a fully convolutional network and a detector. The former proposes regions of interest, while the latter uses the proposed regions (Akok et al., 2017). Faster R-CNN detects objects in two stages: First, it creates a feature map using its convolutional neural network backbone, and then the RPN uses anchor boxes to generate proposed regions for the objects. Second, the feature map is resized for the fully connected layers to classify the object (Tanatipuknon et al., 2021). On the other hand, Mask R-CNN builds on Faster R-CNN with a small overhead, for instance, image segmentation (He et al., 2017). This means that Mask R-CNN can generate a bounding box for a detected object and still predict the object masks in an image. Mask R-CNN architecture comprises a backbone network for feature extraction, an RPN for generating object regions of interest, and two branches for bounding box detection and mask prediction (Sapkota et al., 2024).

Two-stage detectors are highly accurate but generally slow and require substantial computational resources. They have been used with different backbones to detect various objects, such as fruits (Sa et al., 2016; Sapkota et al., 2024), oil palm trees (Zheng et al., 2019), and landslides (Tanatipuknon et al., 2021).

2.5. Model training and application

Our analysis was divided into five steps (1) training the YOLOv3, RetinaNet, Faster R-CNN, and Mask R-CNN algorithms on the initial training dataset (4859 annotations), (2) assessment and comparison of YOLOv3, RetinaNet, Faster R-CNN, and Mask R-CNN performance based on our test dataset (1184 annotations) to select the best model, (3) retraining a new model based on the best model/architecture from step two with 5002 (initial 4859 and new 143) annotations, (4) large-scale detection of tree crowns across the entire study area based on the model from step three, and (5) creation of a georeferenced urban tree height inventory database using extracted LiDAR height metrics (Fig. 3; Table 1).

Step 1: Training the YOLOv3, RetinaNet, Faster R-CNN, and Mask R-CNN models on our initial training dataset (4859 annotations)

A preliminary assessment of the four models was performed to choose the best tree crown detection algorithm for our study area. To achieve this, we trained the four models identically. The initial training data (orthomosaic tiles with corresponding training annotations, Table 1) were divided into 512 x 512 image chips with a stride of 128 x 128 (25 % overlap) to ensure the edge trees were captured. This was necessary to reduce memory requirements, resulting in 31,556 image chips and 102,578 features (associated annotations). The models were trained in ArcGIS Pro 3.3.1 using the deep learning geoprocessing tools with a batch size of 4 and up to 20 epochs (the number of passes through the training data). We applied the default data augmentation to each model to increase training data and improve the robustness. These include cropping, brightness, contrast, dihedral affine, and zooming. The models were trained with consistent parameters for comparability. We used ResNet-50 as the backbone for RetinaNet, Mask R-CNN, and Faster R-CNN, while the default backbone for YOLOv3 was Darknet-53. The software randomly selected 90 % of the training dataset (4859 annotations) for training and 10 % for validation. It also automatically extracted the learning rates, and the weights of the pre-trained backbones were unfrozen. Unfreezing alters backbone parameters and aligns them with the training samples for better results (Hao et al., 2021; Yao et al., 2024). Freezing the backbone weights did not improve model performance, especially for YOLOv3 and Faster R-CCN. We used the validation loss as the metric to monitor overfitting, and the model stopped training when the validation loss did not improve for five epochs. A confidence score threshold of 0.1– 0.5 was tested to select the best value for all the models. A non-maximum suppression (NMS) value of 0.1 was used to remove redundant and overlapping predicted tree crowns. The preliminary model training was conducted on a desktop computer with an Intel(R) Core (TM) i7-8700 CPU and Nvidia GeForce GTX 1060 6 GB, with 32 GB RAM.

Step 2: Assessment and comparison of YOLOv3, RetinaNet, Faster R-CNN, and Mask R-CNN performance based on our test (1184 annotations) to select the best model

The models' outputs (from step 1) were assessed using the test dataset (one orthomosaic tile with 1184 annotations). We compared the models' performance by calculating evaluation metrics (i.e., F1 score, overall accuracy, etc., see below) and visual assessments of

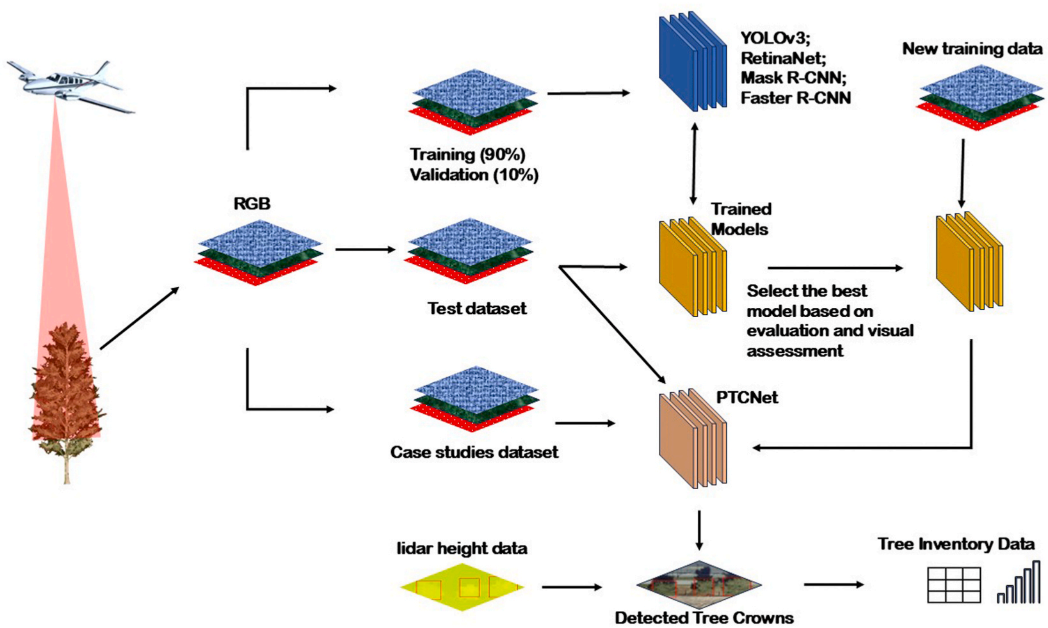

Fig. 3. Schematic diagram of the tree crown detection workflow employed in this study, including the development and assessment of several deep learning models, the selection of the best model, and the final established tree height and location inventory database.

the detected tree crowns.

2.6. Model evaluation

Intersection over union (IoU), box precision, box recall, and F1 scores were used to evaluate the models with the test data. Intersection over union (IoU) is a value ranging from 0 to 1 (Eq. (1)) that measures the average overlap between predicted tree crowns and test data (hand annotations). The IoU is a general indicator of model performance, but it is insufficient for model evaluation. A detected box is deemed accurate if its intersection over the union (IoU) with the hand annotations is ≥ 0.5. This is more stringent and ensures high bounding boxes are selected. The box recall (%; Eq. (2)) is the proportion of true positives that match ground truth based on the used intersection-over-union threshold (0.5). Box precision (%; Eq. (3)) refers to the ratio of the correctly predicted boxes to the total predicted boxes based on the intersection-over-union threshold. The F1 score (Eq. (4)) harmonizes the precision and recall performance. We selected the best model based on a visual assessment and screening of all these metrics, especially the F1 score.

$$\mathrm{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}} \text{ x } 100 \qquad \text{(Equation 1)}$$

$$\text{Box Recall } (\%) = \frac{\text{Correctly Predicted Box (TP)}}{\text{Ground Truth Data (TP + FN)}} \text{ x } 100 \qquad \text{(Equation 2)}$$

$$\text{Box Precision } (\%) = \frac{\text{Correctly Predicted Box (TP)}}{\text{Total Predicted Box (TP + FP)}} \text{ x } 100 \qquad \text{(Equation 3)}$$

$$\text{F1} - \text{Score} = 2 \text{ x } \frac{\text{Precision x Recall}}{\text{Precision + Recall}} \text{ x } 100 \qquad \text{(Equation 4)}$$

Step 3: Retraining a new model based on the best model/architecture from Stage 2, with 5002 annotations to develop a final model

The best model from the preliminary assessment in Step 2 was selected and trained as the final model. Because we wanted an optimal model for detecting trees in Pullman and to further improve the model's learning and performance, we added new annotations (143) from the same area to the initial training annotation (4,859) to retrain a new model (final model). That is, we trained a final model for our study area with 5002 training annotations (Table 1) with some modifications to the processes in Step 2. We used a batch size of 8 instead of 4 in Stage 1 to improve the processing and a maximum of 30 epochs. We also trained for up to 50 epochs separately to assess whether performance would change, but there was no significant improvement. The final model was trained on a desktop computer with a 12th Gen Intel(R) Core (TM) i9-12900K processor and an Nvidia GeForce RTX 3070 Ti GPU, with 128 GB of RAM. Finally, we tested the performance of the final model for two case studies. That is, two selected orthomosaic tiles, one with random tree arrangements within an urban setting with many non-tree features and one tile with clustered tree arrangements (i.e., a high tree density case) (Table 2).

Step 4: Large-scale detection of tree crowns based on the final model from step three across the entire study area

The final model was used to detect tree crowns on a large scale across the study area using the remaining 78 orthomosaic tiles (excluding the single training and seven testing tiles, Fig. 2). The large-scale detected tree crowns were also visually assessed.

Step 5: Creation of a georeferenced urban tree height inventory database using extracted LiDAR height metrics

2.7. LiDAR-based and field survey height data

An airborne LiDAR dataset collected over Whitman County (the County of our study location) in 2018 under the United States Geological Survey (USGS) 3D Elevation Program (3DEP) was used to estimate tree height (OCM Partners, 2024). The Whitman 3DEP 2018 was part of a larger USGS project, in collaboration with the Federal Emergency Management Agency (FEMA) and the Natural Resources Conservation Service (NRCS). The vendor used a Leica ALS80 and a Riegl VQ-1560i sensor system mounted in a Cessna Caravan (OCM Partners, 2024). The average pulse density of the quality one LiDAR data was $\geq$8 pulses/m^2, collected at an altitude of 2300 m with an average relative vertical accuracy of 0.0036 m. We downloaded the DSM and digital terrain model (DTM) datasets from the LiDAR portal (https://lidarportal.dnr.wa.gov/, accessed on 1/22/2025) and calculated the normalized digital surface model

Table 2
Summary statistics of the ground-based tree inventory data ($n = 100$ trees).

Variables	Minimum	Mean	Maximum	Standard Deviation
Diameter (cm)	17	51.9	94.5	16.2
Height (m)	6.6	16.9	27.3	4.7

(nDSM) as the height data. After large-scale tree prediction, the tree crowns (bounding boxes) were used to extract height information from the nDSM. We estimated height metrics (90th, 95th, 98th, and 99th percentiles) based on the nDSM cells within each tree crown using the "Zonal Statistics As Table" tool in ArcGIS Pro 3.3.1 (Hao et al., 2021).

We randomly selected 100 trees on the Washington State University campus in the study area and collected tree inventory data manually in December 2024 to evaluate the accuracy of the nDSM. We collected the coordinates (longitude and latitude) using a handheld Trimble Geo7x device with a positional accuracy of <60 cm, tree height with a TruPulse laser rangefinder, and diameter with a diameter tape. The measured heights were grouped into short (n = 33), medium (n = 33), and tall trees (n = 34) with respective height ranges of 6.6–13.5m, 13.6–18m, and 18.1–27.3m. The summary statistics for the inventory data are shown in Table 2.

The coefficient of determination (R^2) was used to determine the best height metric to attribute to detected trees. We validated the accuracy of the nDSM height values by correlating different LiDAR-derived height metrics (mean, median, 95th, 98th, and 99th height percentiles) with field-based height measurements. The height metric with the highest R^2 value was selected as the height value for the detected trees.

The tree crown tip is expected to be the tallest within the tree crown bounding box; therefore, we filtered the tree inventory data to exclude tree crowns with a maximum height <3 m. This removed dwarf trees, saplings, or newly planted trees.

3. Results

We compared the performance of one-stage (YOLOv3 and RetinaNet) and two-stage (Faster R-CNN and Mask R-CNN) object detection models to select the best model for our study area (Fig. 4). Table 3 shows the evaluation metrics for the models in detecting trees after training and testing, using 4856 and 1184 hand annotations. YOLOv3 achieved the highest F1 score (69 %) and 78 % accuracy in its detected trees. Faster-RCNN had the lowest performance, with an F1 score of 54 %. Confidence scores are values (0–1) assigned by the models that indicate the reliability of the predicted bounding box, and any prediction with a lower score is discarded. We defined a low confidence score as predicted tree crowns with a confidence score of ≥0.1 and <0.3, and a high confidence score as > 0.3. Using visual assessment and F1 scores, we tested different confidence score thresholds across all models, and the best thresholds for YOLOv3, RetinaNet, Faster R-CNN, and Mask R-CNN were 0.1, 0.3, 0.4, and 0.5, respectively. Therefore, the YOLOv3-predicted tree crowns with low confidence scores were mostly accurate (Fig. 5).

Based on the preliminary comparison of the models, YOLOv3 performed best and was chosen as the tree detection model for the study area. A new YOLOv3 (hereafter referred to as PTCNet: Pullman Tree Crowns Network) architecture was trained using 5002 hand-annotated tree crowns for 30 and 50 epochs. Table 4 presents the model's performance on our test datasets (1184 annotations) at both epochs. The test dataset generally represents an urban setting, and the model accurately detected the tree crowns (Fig. 6: A).

The model trained for 30 epochs was selected as the final model based on its F1 score value (harmonic mean of precision and recall) and visual assessment. It was tested further under two different tree arrangement case studies – random and clustered tree arrangement (Table 4; Fig. 6: B and C). PTCNet accurately predicted >80 % of its delineated tree crown boxes in the scenarios, achieving the highest F1 score in the random tree arrangement case study. The model predicted trees with varying crown shapes, closer or farther away from structures, with a precision of 83 %, and trees in a closed canopy, with a precision of 90 % (Table 4). However, a few tree crowns were undetected in the clustered tree scenario (Fig. 6: B).

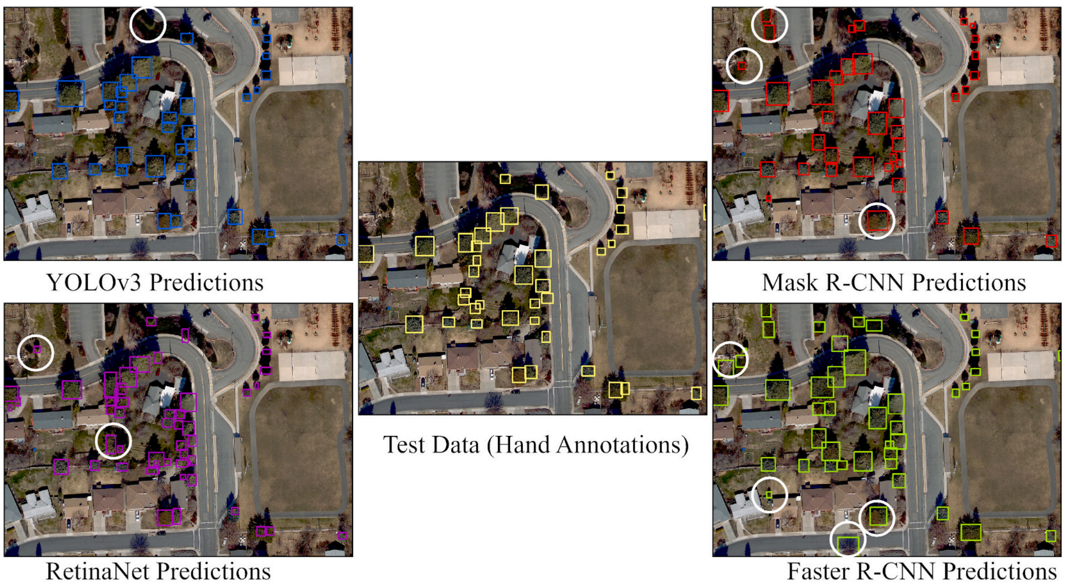

Fig. 4. Comparison of detected tree crowns by the Mask R-CNN, Faster R-CNN, RetinaNet, and YOLOv3 models with the test data (False detections and omitted trees are highlighted in circles).

Table 3
The model evaluation results based on precision, recall, F1-score, and individual best confidence score thresholds. The best confidence score was selected by comparing the metrics based on 0.1–0.5 confidence scores at an IOU of 0.5 and a non-maximum suppression of 0.1.

Model	Precision (%)	Recall (%)	F1 Score (%)	Confidence Score
YOLOv3	**78**	**61**	**69**	**0.1**
RetinaNet	51	64	57	0.3
Mask R-CNN	59	60	60	0.5
Faster R-CNN	56	53	54	0.4

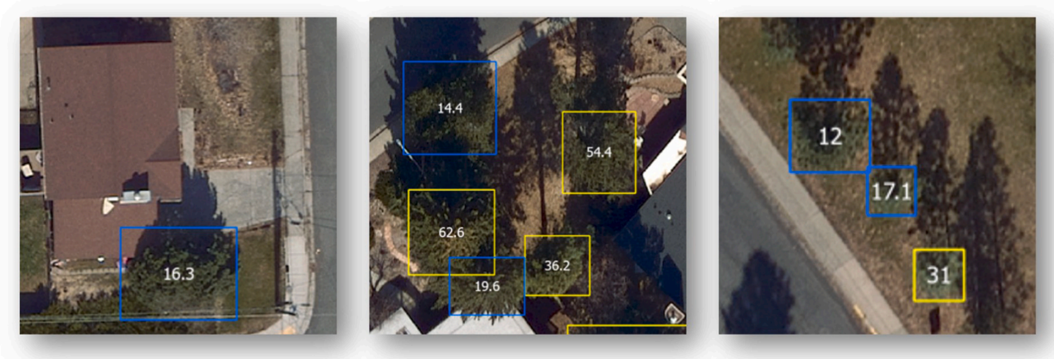

Fig. 5. Detected tree crowns by PTCNet with low confidence and high confidence scores. Confidence scores are values assigned by the models indicating the reliability of the predicted bounding box, and any prediction with lower values will be discarded. We defined low confidence as predicted tree crowns with a confidence score of $\geq$10 and $<$ 30, while a high confidence score was $>$30. Therefore, the detected trees with low confidence (in blue) were accurate as the high-confidence (yellow) tree crowns.

Table 4
Evaluation metrics of the final model based on the YOLOv3 architecture with an intersection-over-union value of 0.5 when tested under different scenarios.

Model	Precision (%)	Recall (%)	F1 Score (%)	Epochs	Location
PTCNet	78	62	69	30	Test area
	81	59	68	50	Test area
	83	72	77		RA
	90	60	72		CA

RA: Trees with random arrangements and close to buildings, CA: High-dense tree canopy arrangement.

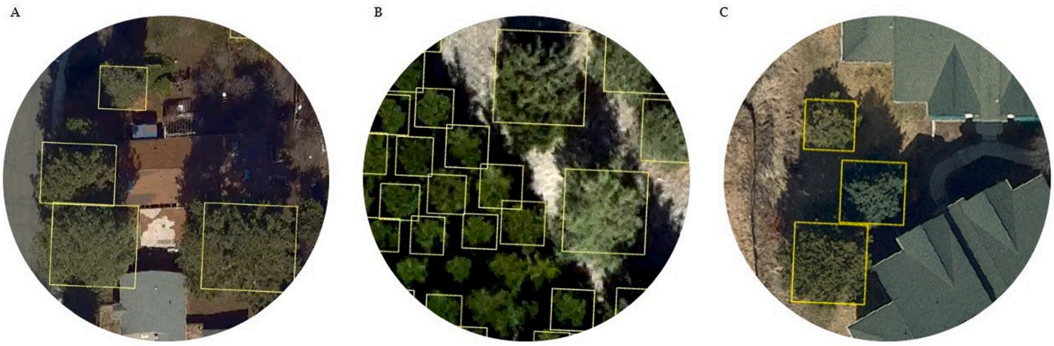

Fig. 6. Examples of detected tree crowns in Pullman under the (A) test dataset and different case studies: (B) high-dense tree canopy arrangement and (C) random tree arrangement close to buildings. A few trees were not detected in the clustered tree arrangement case study.

Fig. 7 shows the regression plots of measured height and LiDAR-based height metrics, grouped by height class. The height percentiles correlated strongly with the measured heights, and we selected the 98th height percentile as the best. The relationship between the measured height (short, medium, and tall classes) and the selected height metric (98th percentile) is shown in Figure A1. The 98th

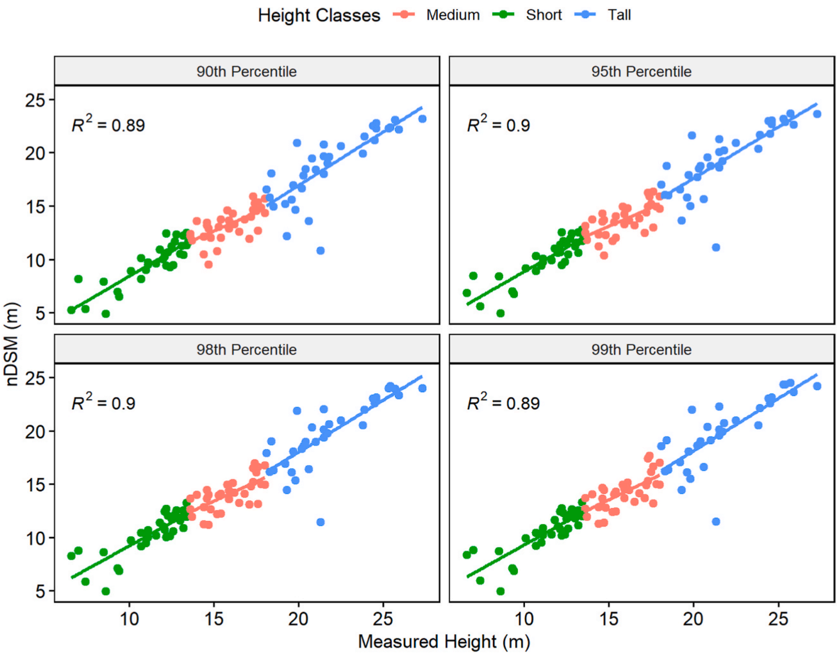

Fig. 7. Regression plot of measured height with lidar-based height metrics (n = 100 trees).

percentile height predicted short trees better ($r^2 = 0.70$), followed by the tall ($R^2 = 0.62$) and medium ($R^2 = 0.46$) trees.

Fig. 8 shows the graphical distribution of tree height (≥3m) based on the 98th percentile height metric within the study area. Trees with height values of >3m to ≤6m had the highest frequency distribution, while tall trees (>33m) had the lowest distribution. The study area was predominantly characterized by trees with heights spanning 6m–21m.

4. Discussion

This study focused on developing a repeatable, straightforward framework for tree crown detection to derive an urban tree inventory in Pullman City using deep learning object detection models, high-resolution images, and LiDAR-based height data. To achieve this, we compared the performance of four CNN-based object detection models and developed PTCNet. This discussion section is divided into two parts: First, we discussed the preliminary assessment of the four CNN models that led to the development of PTCNet. Second, we discussed PTCNet, its application, and limitations.

4.1. Model comparison

Compared to one-stage object detection models, two-stage models are believed and known to achieve higher accuracy because of

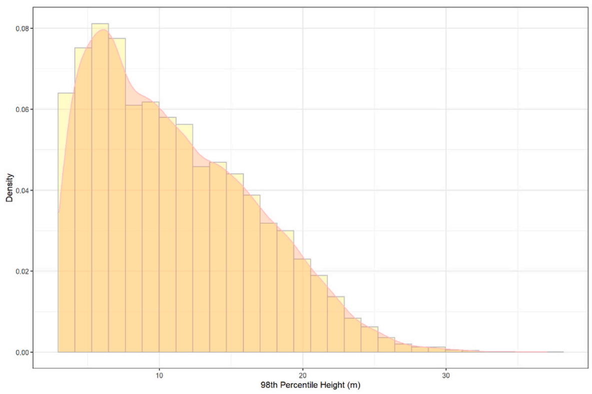

Fig. 8. Height (98th percentile) distribution of the ~12,000 detected tree crowns in Pullman. The evaluation distribution is based on trees with an estimated height of ≥ 3m; i.e., tree crowns with a maximum height of <3m were removed.

their two-stage approach to detecting targets (Lin et al., 2018). For example, Mask R-CNN is regarded as a modern, robust architecture, capable of detecting and predicting an object mask (Yao et al., 2024). However, based on our findings, Mask R-CNN was the second-best model (after YOLOv3), with an F1 score of 60 %, followed by RetinaNet (F1 score: 57 %) and Faster R-CNN, with the least F1 score (54 %). Given the limited studies on Mask R-CNN and Faster R-CNN for urban tree detection, as most of the studies modified the architectures (Zhang et al., 2022) or focused on a single species (dos Santos et al., 2019; Yao et al., 2024), forest trees (Braga et al. 2020), or urban trees using geo-tagged street-level images rather than high-resolution images (Lumnitz et al., 2021), this makes it challenging for a direct comparison. However, some studies have reported that YOLOv3 outperformed Faster R-CNN (Chen et al., 2021; Srivastava et al., 2021; Sharma et al., 2024).

YOLOv3 performance surpassed that of RetinaNet, a similar one-stage object detector that uses a focal loss function to reduce class imbalance during training. This is similar to that of Xia et al. (2019), who reported that YOLOv3 achieved better results than RetinaNet in detecting oil palm trees. Although the RetinaNet backbone (ResNet50) used in this study has a deeper architecture for high accuracy, surprisingly, it produced the highest false-positive tree-crown predictions among all the models. This is slightly different from the findings of dos Santos et al. (2019) where RetinaNet was reported to outperform YOLOv3 and Faster R-CNN. The discrepancy is mainly because our models were trained to detect all species against a single species in their study. Furthermore, our training parameters differ, including the number of epochs and the number of training image tiles.

While all the models struggled with the complexity and noise of the urban environment in RGB images, the YOLOv3 architecture seemed "conservative", predicting a smaller number of tree crowns with high precision. This made it more precise and superior to Mask R-CNN, Faster R-CNN, and RetinaNet, which had more than 40 % of their predicted tree crowns classified as false positives (commission errors). Given that the architecture of the R-CNN family is more complex than that of YOLOv3 and RetinaNet, the training dataset may not be sufficient for the models to generalize accurately in our context. While manual delineation of annotations is time-consuming and laborious, semi-supervision (i.e., including both labeled and unlabeled data into train models) is usually adopted to generate a large number of LiDAR-based annotations (Weinstein et al., 2019). However, this method might not be advantageous in an urban area with buildings and structures of similar tree heights. Additionally, Mask R-CNN and Faster R-CNN require extensive computational resources and might not be feasible for a real-time application (Sapkota et al., 2024).

4.2. PTCNet – urban tree crown detector

PTCNet is a deep-learning urban tree crown detector for our study area, developed on YOLOv3 architecture with a Darknet-53 backbone. Based on the test dataset, the model achieved >75 % precision and 62 % recall on the ground-truth data. However, based on other case study test data, the performance was higher (Table 4). The PTCNet architecture (YOLOv3) was chosen for its high performance among the considered object detection models and for its stable, computationally efficient architecture for tree detection. Although not significantly higher, the final YOLOv3 (PTCNet) model had higher recall than the initial YOLOv3 model. We believe that the additional annotations improved the PTCNet's robustness and performance for the large-scale tree detection.

Two of the significant benefits of PTCNet are (1) the simple architectural workflow and performance in detecting trees of varying

Fig. 9. Examples of small tree canopies detected in the study area based on the large-scale prediction using the PTCNet. The highlighted area (red rectangle) indicates small trees (<3m) with small tree crowns.

canopy structure, and (2) the speed at which inferences are made over new datasets (Redmon and Farhadi, 2018; Tan et al., 2021). Although the YOLOv3 architecture has been reported to have less accuracy in small object detection (i.e., small tree crowns in our case; Liu et al., 2023), we believe that the use of high-spatial-resolution images, data augmentation (such as cropping, brightness, etc.), and the representative sampling of different tree sizes in our study area cushioned this effect (Kaur and Singh, 2023) (Fig. 9). The developed model (PTCNet) was applied on a large scale to predict tree crowns across the city. Precisely, 78 (Fig. 2) high-resolution aerial image tiles (1 km × 1 km) with trees were predicted using PTCNet, and tree crowns were detected across the city with high accuracy. This study presents a useable urban tree crown detector (https://github.com/Okikiola-Michael/PTCNet) for Pullman City. Also, a comprehensive manual and model-generated annotations with height information were developed. The tree crown data generated from this study provides a baseline inventory dataset useful for urban greening and planning, tree monitoring, and the study of climate change impact in an urban environment.

4.2.1. *PTCNet robustness and limitations*

CNN architectures can learn to delineate objects of interest from the training datasets, thereby improving the transferability of a pre-trained model (Weinstein et al., 2019; Chadwick et al., 2024). PTCNet was tested using different tree distribution scenarios to capture its generalization ability and limitations. We tested it on images with random tree arrangements near buildings and clustered trees, as found in forests. Testing the model using these case studies is more rigorous and reveals its weaknesses. In the case studies, PTCNet F1 scores and precision values were >70 % and >80 %, respectively. This demonstrates PTCNet's ability to accurately detect tree crowns with various spatial arrangements. Although the model's performance is high according to the evaluation metrics, not all trees were detected, highlighting the limitations of the PTCNet architecture. Also, our single-site training approach may limit the model's generalization when applied to another study area. Therefore, further retraining may be needed to improve the model's performance in other study locations.

4.2.2. *Height estimation*

Aerial LiDAR data have been consistently used to measure tree heights and are found to be accurate (Andersen et al., 2006). We extracted different height metrics for the predicted tree crowns across the city using nDSM data. The georeferenced tree crowns (bounding boxes) with the height metrics are available on a dedicated GitHub repository (https://github.com/Okikiola-Michael/PTCNet). The 98th height percentile distribution showed that most trees were taller than 3m and less than 30m. This is particularly relevant for land and city managers, as metrics related to species structure and composition, such as vegetation height, are mostly preferred (Meddens et al., 2022). We suspected that the average height within a tree crown box would be biased because of the spaces in the tree crown bounding boxes, making the mean value to be affected by outliers. Though the LiDAR data was collected in 2018, the height data, the first of its kind, offers insight into the tree species diversity within Pullman. Tree vertical growth depends on several factors, which may or may not be significant over five years (the time difference between the LiDAR and image data); this may have led to an underestimation of tree heights in the study area. Also, there might have been changes in the number and distribution of trees within the city. Nonetheless, the selected height metric (98th percentile) showed a strong correlation with the field-surveyed height, particularly for both short and tall trees. Thus, it provides a baseline for other studies that focus on extracting large-scale relevant inventory data. Additionally, future studies can use our data to support tree species mapping and widen the scope of the tree inventory.

4.2.3. *Limitations and future research*

One limitation of this study is the imagery used, which was collected in March, when some trees lacked foliage. First, this limits the total number of trees detected in the study area. However, in subsequent analysis, NAIP imagery can be used to train a new model, or PTCNet can be retrained on representative NAIP imagery samples from the study area. Therefore, more comparative analysis is needed to assess differences in tree detection in a complex urban setting like ours, using aerial images with a spatial resolution of 8 cm and NAIP imagery of ~0.5–1m. Second, the images included noise from structures and buildings, as well as grass and shrubs, such as ornamental cypress shrubs (*Microbiota decussata*) (Figure A2). Although we attempted to mitigate the impact of this noise through careful image annotation and interpretation by an expert with local knowledge, some of these errors may have affected the model's performance, leading to false-positive detections. Future models can be trained to identify the noise, especially the cypress, as a separate class to improve the model's performance efficiently.

The use of YOLOv3 may have limited the model's performance relative to the current state-of-the-art YOLO variants. Although the goal was to develop a simple, repeatable workflow with a final model useable by anyone (especially city managers) and supported by available computational resources, newer versions of YOLO with more advanced architectures could increase the number of correctly detected trees, thereby improving PTCNet performance. For example, YOLOv7 has been reported to have the highest accuracy and speed among the YOLO series (Xu et al., 2024). However, this is not yet supported in ArcGIS Pro, and the performance of YOLOv7 needs to be investigated in our study area. The current version of PTCNet is operational, has been deployed on a large scale, and serves as the initial step in monitoring urban trees.

5. Conclusion

This study tested four object detection models – YOLOv3, RetinaNet, Mask R-CNN, and Faster R-CNN for urban tree detection, and our results showed that YOLOv3 outperformed the other models. The YOLOv3 architecture was used to develop a final model (PTCNet), which was applied at a large scale to detect ~12,000 trees in the study area. Further analysis showed that our study area contains trees with varying height ranges. However, the tree heights might have been underestimated due to temporal differences between the image, LiDAR, and field survey data. However, the 98th percentile height metric correlated strongly with the field measurements.

This study developed the first large-scale tree crown detection in our study area through a comparative analysis of advanced object detection models. We presented findings from the performance of four state-of-the-art deep learning models in detecting trees in a complex urban setting, and developed datasets that will support urban planning, greening, and tree-related studies at various scales.

CRediT authorship contribution statement

Okikiola Michael Alegbeleye: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Formal analysis, Data curation, Conceptualization. **Arjan Johan Herman Meddens:** Writing – review & editing, Supervision, Funding acquisition, Data curation, Conceptualization. **Yetunde Oladepe Rotimi:** Writing – review & editing, Validation, Methodology. **Kelechi Godwin Ibeh:** Writing – review & editing, Validation.

Ethical statement

All the authors declare that ethical practices have been followed in relation to the development, writing, and publication of the article.

Funding source

The USDA National Institute of Food and Agriculture (NIFA), McIntire Stennis Project 1019284, funded this work.

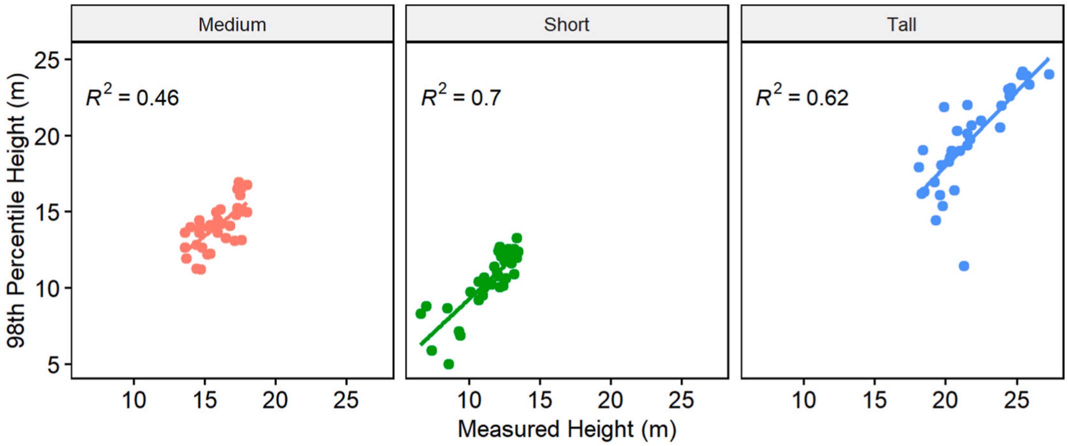

Fig. A1. Regression plot of measured height with the selected height metric (98th percentile) based on height classes (n = 100 trees).

Fig. A2. Examples of challenges in the tree crown detection within the imagery.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We thank the reviewers for helpful comments and Sam Jenkins and the City of Pullman for sharing the aerial imagery and contributions to earlier drafts of this manuscript.

Data availability

The geospatial file of the detected trees with associated height values, the final model, and other datasets are publicly available on our GitHub page: https://github.com/Okikiola-Michael/PTCNet.

References

Akok, B., Gurkan, F., Kaplan, O., Gunsel, B., 2017. Robust object tracking by interleaving variable rate color particle filtering and deep learning. In: 2017 IEEE International Conference on Image Processing (ICIP), pp. 3665–3669. https://doi.org/10.1109/ICIP.2017.8296966.

Andersen, H.-E., Reutebuch, S.E., McGaughey, R.J., 2006. A rigorous assessment of tree height measurements obtained using airborne LIDAR and conventional field methods. Can. J. Rem. Sens. 32 (5), 355–366. Available from. https://research.fs.usda.gov/treesearch/25922. (Accessed 20 November 2024).

Ang, G., Zhiwei, T., Wei, M., Yuepeng, S., Longlong, R., Yuliang, F., Jianping, Q., Lijia, X., 2024. Fruits hidden by green: an improved YOLOV8n for detection of young citrus in lush citrus trees. Front. Plant Sci. 15. https://doi.org/10.3389/fpls.2024.1375118. Frontiers.

Bao, W., Fan, T., Hu, G., Liang, D., Li, H., 2022. Detection and identification of tea leaf diseases based on AX-RetinaNet. Sci. Rep. 12 (1), 2183. https://doi.org/10.1038/s41598-022-06181-z. Nature Publishing Group.

Bechtold, W.A., Patterson, P.L. (Eds.), 2005. The Enhanced Forest Inventory and Analysis Program - National Sampling Design and Estimation Procedures, vol. 85. U. S. Department of Agriculture, Forest Service, Southern Research Station, Asheville, NC, p. 80. https://doi.org/10.2737/SRS-GTR-80. Gen. Tech. Rep. SRS-80.

Beloiu, M., Heinzmann, L., Rehush, N., Gessler, A., Griess, V.C., 2023. Individual tree-crown detection and species identification in heterogeneous forests using aerial RGB imagery and deep learning. Remote Sens. 15 (5), 1463. https://doi.org/10.3390/rs15051463. Multidisciplinary Digital Publishing Institute.

Braga, J.R.G., Peripato, V., Dalagnol, R., Ferreira, M.P., Tarabalka, Y.O.C., Aragão, L.E., de Campos Velho, H.F., Shiguemori, E.H., Wagner, F.H., 2020a. Tree crown delineation algorithm based on a convolutional neural network. Remote Sens. 12 (8). https://doi.org/10.3390/rs12081288.

Braga, J.R.G., Peripato, V., Dalagnol, R. P. Ferreira M., Tarabalka, Y.O.C., Aragão, L.E., de Campos Velho, H.F., Shiguemori, E.H., Wagner, F.H., 2020b. Tree crown delineation algorithm based on a convolutional neural network. Remote Sens. 12 (8), 1288. https://doi.org/10.3390/rs12081288. Multidisciplinary Digital Publishing Institute.

Brandtberg, T., 1999. Automatic individual tree based analysis of high spatial resolution aerial images on naturally regenerated boreal forests. Can. J. For. Res. 29 (10), 1464–1478. https://doi.org/10.1139/x99-150. NRC Research Press.

Chadwick, A.J., Coops, N.C., Bater, C.W., Martens, L.A., White, B., 2024. Transferability of a Mask R–CNN model for the delineation and classification of two species of regenerating tree crowns to untrained sites. Science of Remote Sensing 9, 100109. https://doi.org/10.1016/j.srs.2023.100109.

Chen, L., Shi, W., Deng, D., 2021. Improved YOLOv3 based on attention mechanism for fast and accurate ship detection in optical remote sensing images. Remote Sens. 13 (4), 660. https://doi.org/10.3390/rs13040660. Multidisciplinary Digital Publishing Institute.

Choi, K., Lim, W., Chang, B., Jeong, J., Kim, I., Park, C.-R., Ko, D.W., 2022. An automatic approach for tree species detection and profile estimation of urban street trees using deep learning and Google street view images. ISPRS J. Photogrammetry Remote Sens. 190, 165–180. https://doi.org/10.1016/j.isprsjprs.2022.06.004.

Dai, J., Li, Y., He, K., Sun, J., 2023. R-FCN: object detection via region-based fully convolutional networks. arXiv. https://doi.org/10.48550/arXiv.1605.06409.

Delplanque, A., Foucher, S., Lejeune, P., Linchant, J., Théau, J., 2022. Multispecies detection and identification of African mammals in aerial imagery using convolutional neural networks. Remote Sensing in Ecology and Conservation 8 (2), 166–179. https://doi.org/10.1002/rse2.234.

dos Santos, A.A., Marcato Junior, J., Araújo, M.S., Di Martini, D.R., Tetila, E.C., Siqueira, H.L., Aoki, C., Eltner, A., Matsubara, E.T., Pistori, H., Feitosa, R.Q., Liesenberg, V., Gonçalves, W.N., 2019. Assessment of CNN-based methods for individual tree detection on images captured by RGB cameras attached to UAVS. Sensors (Switzerland) 19 (16). https://doi.org/10.3390/s19163595.

Erikson, M., 2003. Segmentation of individual tree crowns in colour aerial photographs using region growing supported by fuzzy rules. Can. J. For. Res. 33 (8), 1557–1563. https://doi.org/10.1139/x03-062. NRC Research Press.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587. https://doi.org/10.1109/CVPR.2014.81.

Gu, J., Grybas, H., Congalton, R.G., 2020. Individual tree crown delineation from UAS imagery based on Region growing and growth space considerations. Remote Sens. 12 (15), 2363. https://doi.org/10.3390/rs12152363. Multidisciplinary Digital Publishing Institute.

Gui, S., Song, S., Qin, R., Tang, Y., 2024. Remote sensing object detection in the deep learning Era—A review. Remote Sens. 16 (2), 327. https://doi.org/10.3390/rs16020327. Multidisciplinary Digital Publishing Institute.

Hao, Z., Lin, L., Post, C.J., Mikhailova, E.A., Li, M., Chen, Y., Yu, K., Liu, J., 2021. Automated tree-crown and height detection in a young forest plantation using mask region-based convolutional neural network (Mask R-CNN). ISPRS J. Photogrammetry Remote Sens. 178, 112–123. https://doi.org/10.1016/j.isprsjprs.2021.06.003.

Hao, Q.S., Chun, H.Y., Yong, Z.S., 2022. A mulberry detection method in the complex background based on improved D-YOLOv3. In: 2022 International Conference on Networks, Communications and Information Technology (CNCIT), pp. 6–13. https://doi.org/10.1109/CNCIT56797.2022.00010.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask R-CNN. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988. https://doi.org/10.1109/ICCV.2017.322.

Hoeser, T., Kuenzer, C., 2020. Object detection and image segmentation with deep learning on Earth observation data: a review-part I: evolution and recent trends. Remote Sens. 12 (10), 1667. https://doi.org/10.3390/rs12101667. Multidisciplinary Digital Publishing Institute.

Hoeser, T., Bachofer, F., Kuenzer, C., 2020. Object detection and image segmentation with deep learning on Earth observation data: a Review—part II: applications. Remote Sens. 12 (18), 3053. https://doi.org/10.3390/rs12183053. Multidisciplinary Digital Publishing Institute.

Huo, L., Lindberg, E., 2020. Individual tree detection using template matching of multiple rasters derived from multispectral airborne laser scanning data. Int. J. Rem. Sens. 41, 9525–9544. https://doi.org/10.1080/01431161.2020.1800127.

Itakura, K., Hosoi, F., 2020. Automatic tree detection from three-dimensional images reconstructed from 360° spherical camera using YOLO v2. Remote Sens. 12 (6), 988. https://doi.org/10.3390/rs12060988. Multidisciplinary Digital Publishing Institute.

Kaur, R., Singh, S., 2023. A comprehensive review of object detection with deep learning. Digit. Signal Process. 132, 103812. https://doi.org/10.1016/j.dsp.2022.103812.

Lamar, W.R., McGraw, J.B., Warner, T.A., 2005. Multitemporal censusing of a population of eastern hemlock (Tsuga canadensis L.) from remotely sensed imagery using an automated segmentation and reconciliation procedure. Rem. Sens. Environ. 94 (1), 133–143. https://doi.org/10.1016/j.rse.2004.09.003.

Li, M., Yao, W., 2020. 3d map system for tree monitoring in Hong Kong using google street view imagery and deep learning. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **V-3–2020**. Copernicus GmbH, pp. 765–772. https://doi.org/10.5194/isprs-annals-V-3-2020-765-2020.

Li, W., Fu, H., Yu, L., Cracknell, A., 2017. Deep learning based oil palm tree detection and counting for high-resolution remote sensing images. Remote Sens. 9 (1), 22. https://doi.org/10.3390/rs9010022. Multidisciplinary Digital Publishing Institute.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2018. Focal loss for dense object detection. arXiv. https://doi.org/10.48550/arXiv.1708.02002.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C., 2016. SSD: single shot MultiBox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), Computer Vision – ECCV 2016. Springer International Publishing, Cham, pp. 21–37. https://doi.org/10.1007/978-3-319-46448-0_2.
Liu, B., He, F., Du, S., Li, J., Liu, W., 2023. An advanced YOLOv3 method for small object detection. arXiv. https://doi.org/10.48550/arXiv.2212.02809.
Liu, Y., Chen, D., Fu, S., Mathiopoulos, P.T., Sui, M., Na, J., Peethambaran, J., 2024. Segmentation of individual tree points by combining marker-controlled watershed segmentation and spectral clustering optimization. Remote Sens. 16 (4), 610. https://doi.org/10.3390/rs16040610. Multidisciplinary Digital Publishing Institute.
Lumnitz, S., Devisscher, T., Mayaud, J.R., Radic, V., Coops, N.C., Griess, V.C., 2021. Mapping trees along urban street networks with deep learning and street-level imagery. ISPRS J. Photogrammetry Remote Sens. 175, 144–157. https://doi.org/10.1016/j.isprsjprs.2021.01.016.
Ma, M., Liu, J., Liu, M., Zeng, J., Li, Y., 2021. Tree species classification based on Sentinel-2 imagery and random Forest classifier in the Eastern regions of the Qilian Mountains. Forests 12 (12), 1736. https://doi.org/10.3390/f12121736. Multidisciplinary Digital Publishing Institute.
Meddens, A.J.H., Steen-Adams, M.M., Hudak, A.T., Mauro, F., Byassee, P.M., Strunk, J., 2022. Specifying geospatial data product characteristics for forest and fuel management applications. Environ. Res. Lett. 17 (4), 045025. https://doi.org/10.1088/1748-9326/ac5ee0. IOP Publishing.
Nurul Husna, S., Abdul Shukor, S., Johari, J., 2019. A review on remote sensing-based method for tree detection and delineation. IOP Conf. Ser. Mater. Sci. Eng. 705, 012024. https://doi.org/10.1088/1757-899X/705/1/012024.
OCM Partners, 2024. 2018 WA DNR Lidar: Chelan, WA from 2010-06-15 to 2010-08-15. NOAA National Centers for Environmental Information. Available from. https://www.fisheries.noaa.gov/inport/item/67339.
Pham, M.-T., Courtrai, L., Friguet, C., Lefèvre, S., Baussard, A., 2020. YOLO-Fine: One-Stage detector of small objects under various backgrounds in remote sensing images. Remote Sens. 12 (15). https://doi.org/10.3390/rs12152501.
Plowright, A.A., Coops, N.C., Eskelson, B.N.I., Sheppard, S.R.J., Aven, N.W., 2016. Assessing urban tree condition using airborne light detection and ranging. Urban For. Urban Green. 19, 140–150. https://doi.org/10.1016/j.ufug.2016.06.026.
Pu, R., 2021. Mapping tree species using advanced remote sensing technologies: a state-of-the-art review and perspective. J Remote Sens 2021, 2021–9812624. https://doi.org/10.34133/2021/9812624.
Qian, C., Yao, C., Ma, H., Xu, J., Wang, J., 2023. Tree species classification using airborne LiDAR data based on individual tree segmentation and shape fitting. Remote Sens. 15 (2), 406. https://doi.org/10.3390/rs15020406. Multidisciplinary Digital Publishing Institute.
Redmon, J., Farhadi, A., 2018. YOLOv3: an incremental improvement. arXiv. https://doi.org/10.48550/arXiv.1804.02767.
Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788. https://doi.org/10.1109/CVPR.2016.91.
Ren, S., He, K., Girshick, R., Sun, J., 2017. Faster R-CNN: towards real-time object detection with Region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. 39 (6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031.
Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., McCool, C., 2016. DeepFruits: a fruit detection System using deep neural networks. Sensors 16 (8), 1222. https://doi.org/10.3390/s16081222. Multidisciplinary Digital Publishing Institute.
Safonova, A., Hamad, Y., Alekhina, A., Kaplun, D., 2022. Detection of Norway spruce trees (Picea abies) infested by bark beetle in UAV images using YOLOs architectures. IEEE Access 10, 10384–10392. https://doi.org/10.1109/ACCESS.2022.3144433.
Sankaran, S., 2024. Multi-Species Object Detection in Drone Imagery for Population Monitoring of Endangered Animals. https://doi.org/10.48550/arXiv.2407.00127 arXiv.
Sapkota, R., Ahmed, D., Karkee, M., 2024. Comparing YOLOv8 and mask RCNN for object segmentation in complex orchard environments. arXiv. https://doi.org/10.48550/arXiv.2312.07935.
Schiefer, F., Kattenborn, T., Frick, A., Frey, J., Schall, P., Koch, B., Schmidtlein, S., 2020. Mapping forest tree species in high resolution UAV-Based RGB-imagery by means of convolutional neural networks. ISPRS J. Photogrammetry Remote Sens. 170, 205–215. https://doi.org/10.1016/j.isprsjprs.2020.10.015.
Sharma, A., Kumar, V., Longchamps, L., 2024. Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and faster R-CNN models for detection of multiple weed species. Smart Agricultural Technology 9, 100648. https://doi.org/10.1016/j.atech.2024.100648.
Skurikhin, A.N., Garrity, S.R., McDowell, N.G., Cai, D.M., 2013. Automated tree crown detection and size estimation using multi-scale analysis of high-resolution satellite imagery. Remote Sensing Letters 4 (5), 465–474. https://doi.org/10.1080/2150704X.2012.749361. Taylor & Francis.
Srivastava, S., Divekar, A.V., Anilkumar, C., Naik, I., Kulkarni, V., Pattabiraman, V., 2021. Comparative analysis of deep learning image detection algorithms. J. Big Data 8 (1), 66. https://doi.org/10.1186/s40537-021-00434-w.
Tan, L., Huangfu, T., Wu, L., Chen, W., 2021. Comparison of RetinaNet, SSD, and YOLO v3 for real-time pill identification. BMC Med. Inf. Decis. Making 21 (1), 324. https://doi.org/10.1186/s12911-021-01691-8.
Tanatipuknon, A., Aimmanee, P., Watanabe, Y., Murata, K.T., Wakai, A., Sato, G., Hung, H.V., Tungpimolrut, K., Keerativittayanun, S., Karnjana, J., 2021. Study on combining two faster R-CNN models for landslide detection with a classification decision tree to improve the detection performance. J. Disaster Res. 16 (4), 588–595. https://doi.org/10.20965/jdr.2021.p0588.
Tian, Y., Yang, G., Wang, Z., Wang, H., Li, E., Liang, Z., 2019. Apple detection during different growth stages in orchards using the improved YOLO-V3 model. Comput. Electron. Agric. 157, 417–426. https://doi.org/10.1016/j.compag.2019.01.012.
Uematsu, R.T., Silva, F.A., Almeida, L.L., Pereira, D.R., Artero, A.O., Piteri, M.A., 2022. Individual tree identification in urban areas from aerial images using mask R-CNN. J. Urban Environ. Eng. 16 (1), 45–54. Journal of Urban and Environmental Engineering.
Velasquez-Camacho, L., Etxegarai, M., de-Miguel, S., 2023. Implementing deep learning algorithms for urban tree detection and geolocation with high-resolution aerial, satellite, and ground-level images. Comput. Environ. Urban Syst. 105, 102025. https://doi.org/10.1016/j.compenvurbsys.2023.102025.
Ventura, J., Pawlak, C., Honsberger, M., Gonsalves, C., Rice, J., Love, N.L.R., Han, S., Nguyen, V., Sugano, K., Doremus, J., Fricker, G.A., Yost, J., Ritter, M., 2024. Individual tree detection in large-scale urban environments using high-resolution multispectral imagery. Int. J. Appl. Earth Obs. Geoinf. 130, 103848. https://doi.org/10.1016/j.jag.2024.103848.
Wagner, F.H., Sanchez, A., Tarabalka, Y., Lotte, R.G., Ferreira, M.P., Aidar, M.P.M., Gloor, E., Phillips, O.L., Aragão, L.E.O.C., 2019. Using the U-net convolutional network to map forest types and disturbance in the Atlantic rainforest with very high resolution images. Remote Sensing in Ecology and Conservation 5 (4), 360–375. https://doi.org/10.1002/rse2.111.
Wang, Y., Hyyppä, J., Liang, X., Kaartinen, H., Yu, X., Lindberg, E., Holmgren, J., Qin, Y., Mallet, C., Ferraz, A., Torabzadeh, H., Morsdorf, F., Zhu, L., Liu, J., Alho, P., 2016. International benchmarking of the individual tree detection methods for modeling 3-D canopy structure for silviculture and forest ecology using airborne laser scanning. IEEE Trans. Geosci. Rem. Sens. 54 (9), 5011–5027. https://doi.org/10.1109/TGRS.2016.2543225.
Wang, J., Zhen, Z., Zhao, Y., Ma, Y., Zhao, Y., 2024. 3D-CNN with multi-scale fusion for tree crown segmentation and species classification. Remote Sens. 16 (23), 4544. https://doi.org/10.3390/rs16234544. Multidisciplinary Digital Publishing Institute.
Weinstein, B.G., Marconi, S., Bohlman, S., Zare, A., White, E., 2019. Individual tree-crown detection in RGB imagery using semi-supervised deep learning neural networks. Remote Sens. 11 (11), 1309. https://doi.org/10.3390/rs11111309. Multidisciplinary Digital Publishing Institute.
Wu, K., Zhang, J., Yin, X., Wen, S., Lan, Y., 2023. An improved YOLO model for detecting trees suffering from pine wilt disease at different stages of infection. Remote Sensing Letters 14 (2), 114–123. https://doi.org/10.1080/2150704X.2022.2161843. Taylor & Francis.
Xi, Z., Hopkinson, C., Rood, S.B., Peddle, D.R., 2020. See the forest and the trees: effective machine and deep learning algorithms for wood filtering and tree species classification from terrestrial laser scanning. ISPRS J. Photogrammetry Remote Sens. 168, 1–16. https://doi.org/10.1016/j.isprsjprs.2020.08.001.
Xia, M., Li, W., Fu, H., Yu, L., Dong, R., Zheng, J., 2019. Fast and robust detection of oil palm trees using high-resolution remote sensing images. In: Automatic Target Recognition XXIX. SPIE, pp. 65–73. https://doi.org/10.1117/12.2518352.
Xia, K., Wang, H., Yang, Y., Du, X., Feng, H., 2021. Automatic detection and parameter estimation of Ginkgo biloba in urban environment based on RGB images. J. Sens. 2021 (1), 6668934. https://doi.org/10.1155/2021/6668934.
Xu, S., Wang, R., Shi, W., Wang, X., 2024. Classification of tree species in transmission line corridors based on YOLO v7. Forests 15 (1), 61. https://doi.org/10.3390/f15010061. Multidisciplinary Digital Publishing Institute.

Yang, M., Mou, Y., Liu, S., Meng, Y., Liu, Z., Li, P., Xiang, W., Zhou, X., Peng, C., 2022. Detecting and mapping tree crowns based on convolutional neural network and google Earth images. Int. J. Appl. Earth Obs. Geoinf. 108, 102764. https://doi.org/10.1016/j.jag.2022.102764.

Yao, S., Hao, Z., Post, C.J., Mikhailova, E.A., Lin, L., 2024. Individual tree crown detection and classification of live and dead trees using a mask region-based convolutional neural network (mask R-CNN). Forests 15 (11), 1900. https://doi.org/10.3390/f15111900. Multidisciplinary Digital Publishing Institute.

Yinghua, Z., Yu, T., Xingxing, L., 2021. Urban street tree recognition method based on machine vision. In: 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), pp. 967–971. https://doi.org/10.1109/ICSP51882.2021.9408928.

ZamboniThgeThe, P.A.P., Marcato, J., Miyoshi, G.T., de Andrade Silva, J., Martins, J., Gonçalves, W.N., 2021. Assessment of CNN-based methods for single tree detection on high-resolution RGB images in urban areas. In: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, pp. 590–593. https://doi.org/10.1109/IGARSS47720.2021.9553092.

Zhang, L., Lin, H., Wang, F., 2022. Individual tree detection based on high-resolution RGB images for urban forestry applications. IEEE Access 10, 46589–46598. https://doi.org/10.1109/ACCESS.2022.3171585.

Zhang, Y., Wang, M., Mango, J., Xin, L., Meng, C., Li, X., 2024. Individual tree detection and counting based on high-resolution imagery and the canopy height model data. Geo-Spatial Inf. Sci. 27 (6), 2162–2178. https://doi.org/10.1080/10095020.2023.2299146. Taylor & Francis.

Zhao, H., Morgenroth, J., Pearse, G., Schindler, J., 2023. A systematic review of individual tree crown detection and delineation with convolutional neural networks (CNN). Curr. For. Rep. 9 (3), 149–170. https://doi.org/10.1007/s40725-023-00184-3. Springer Int Publ Ag, Cham.

Zheng, Y., Wu, G., 2021. Single shot MultiBox detector for urban plantation single tree detection and location with high-resolution remote sensing imagery. Front. Environ. Sci. 9. https://doi.org/10.3389/fenvs.2021.755587. Frontiers.

Zheng, J., Li, W., Xia, M., Dong, R., Fu, H., Yuan, S., 2019. Large-scale oil palm tree detection from high-resolution remote sensing images using Faster-RCNN. In: IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, pp. 1422–1425. https://doi.org/10.1109/IGARSS.2019.8898360.

Zhou, H., Wu, S., Xu, Z., Sun, H., 2024. Automatic detection of standing dead trees based on improved YOLOv7 from airborne remote sensing imagery. Front. Plant Sci. 15, Frontiers. https://doi.org/10.3389/fpls.2024.1278161.

Zhu, Y., Zhou, J., Yang, Y., Liu, L., Liu, F., Kong, W., 2022. Rapid target detection of fruit trees using UAV imaging and improved light YOLOv4 algorithm. Remote Sens. 14 (17), 4324. https://doi.org/10.3390/rs14174324. Multidisciplinary Digital Publishing Institute.