Accurate tropical forest individual tree crown delineation from RGB imagery using Mask R-CNN

Sebastian H. M. Hickman[*a], James G. C. Ball[*b,c], Tobias D. Jackson[*b,c], Xian Jing Koay[a], James Hirst[d], William Jay[e], Mélaine Aubry-Kientz[f], Grégoire Vincent[g], and David A. Coomes[b,c] ✉

[a]Yusuf Hamied Department of Chemistry, University of Cambridge, Lensfield Road, Cambridge, CB2 1EW. United Kingdom.
[b]Department of Plant Sciences, University of Cambridge, Downing Street, Cambridge, CB2 3EA. United Kingdom.
[c]Conservation Research Institute, University of Cambridge, Downing Street, Cambridge, CB2 3EA. United Kingdom.
[d]Department of Applied Mathematics and Theoretical Physics, Wilberforce Road, Cambridge, CB3 0WA, United Kingdom
[e]Plymouth Marine Laboratory, Prospect Place, Plymouth, PL1 3DH. United Kingdom.
[f]AgroParisTech, UMR EcoFoG Kourou
[g]UMR AMAP, University of Montpellier, IRD, CNRS, CIRAD, INRAE, Montpellier, France

[*]**Authors contributed equally to the work (first authors)**

1. Tropical forests are a major component of the global carbon cycle and home to two-thirds of terrestrial species. Upper-canopy trees store the majority of forest carbon and can be particularly vulnerable to drought events and storms. Monitoring their growth and mortality is essential to understanding forest resilience to climate change, but large trees are underrepresented in traditional field surveys, so estimates are poorly constrained.

2. Aerial photographs provide spectral and textural information to discriminate between tree crowns in diverse, complex tropical canopies, potentially opening the door to landscape monitoring of large trees. Here we describe a new deep convolutional neural network machine learning method, *Detectree2*, which builds on the Mask R-CNN computer vision framework to recognise the irregular edges of individual tree crowns from airborne RGB imagery. We trained and evaluated this model with 3,600 manually delineated tree crowns at three sites in Malaysian Borneo and one site in French Guiana. As an example application of this tool, we combined the delineations with repeat lidar surveys of the four sites to estimate the growth and mortality of upper-canopy trees.

3. *Detectree2* delineated 65,000 upper-canopy trees across 14 km^2 of aerial images. The skill of the automatic method in delineating the upper-canopy tree crowns was good (F_1 = 0.71). As predicted from previous field studies, we found that growth rate declined with tree height and tall trees had higher mortality rates than intermediate-size trees. In addition, tall trees in French Guiana had higher growth and mortality rates than those in Borneo.

4. Our approach demonstrates that machine learning methods can automatically segment trees in abundant, cheap RGB imagery. This tool has many potential applications in forest ecology and conservation, from estimating carbon stocks to monitoring forest restoration. We demonstrate its use in tracking the growth and mortality rates of upper-canopy trees at scales much larger than those achievable with field data.

PyTorch based Python package available to install at `https://github.com/PatBall1/Detectree2`

Tropical forests | Tree crown delineation | Tree crown segmentation | Deep learning | Convolutional neural networks | Mask R-CNN | Detectron2 | Tree growth | Tree mortality | Forest monitoring

Correspondence: ***shmh4@cam.ac.uk; ball.jgc@gmail.com***

Introduction

Intact tropical forests are an important component of the global carbon cycle: they are major carbon stores and significant carbon sinks (1). However, the strength of the carbon sink is diminishing as a result of global warming (2, 3) and there are concerns that forests are reaching a tipping point beyond which they could switch irreversibly to open savanna systems (4). However, forecasting the future of tropical forests is challenging, because little is known about the ways different species will respond to changing climate, or the resilience provided by that diversity (5–8). To understand the likely responses of forests to further climate change, ecosystem models need to represent growth and mortality processes of individual trees more accurately than is currently the case (9, 10).

Remote sensing of individual upper-canopy trees can improve estimates of forest carbon (11) and provide a means of tracking growth and mortality over large spatial scales. Traditional monitoring approaches rely on measuring stem dimensions in permanent inventory plots, and periodically revisiting those plots to assess recruitment, growth and mortality (12). However, the coverage of such plots in the tropics is limited (~0.0002% of tropical forests are sampled by the main plot networks) and their locations are dictated by ease of access rather than by robust statistical sampling designs (13–15). Furthermore, upper-canopy trees store the majority of carbon in tropical forests, but few of them are sampled in inventory plots (16–18). This under-sampling is particularly problematic when assessing impacts of climate change, because upper-canopy trees are most vulnerable to periods of water shortage (19, 20) which are increasing in frequency (21). Remote sensing has the potential to overcome these sampling challenges by providing wall-to-wall maps that can be used to monitor millions of upper-canopy trees.

Remote sensing of individual trees has focussed on airborne lidar data, which is used by the forestry industry to map trees at landscape scales (22). Delineating individual trees from airborne lidar datasets is most successful for coniferous forests, because their apical dominance results in clear local height maxima that make tree crowns easily distinguishable (11, 23), but complex tropical canopies have presented a far

DRAFT

greater challenge for lidar delineation (24). Tropical forest canopies are often densely packed with partially interwoven trees which point-cloud clustering algorithms can struggle to distinguish (25). Furthermore, airplanes or helicopters are required to conduct lidar surveys whereas standard RGB imagery is far cheaper and easier to collect because of the availability of low-cost drones.

Automatic delineation of trees in RGB photographs can harness colour and texture information to distinguish trees, even if they are structurally similar (26, 27). Most current methods of individual tree identification from RGB imagery use bounding boxes (Fig. S3) (28–30), but more exact delineation of the edges of tree crowns would provide information on crown area and lateral growth, and avoid mixing of signals from neighbouring trees. Recent advances in neural network approaches to computer vision provide opportunities to recognise individual trees from standard digital photographs taken from drones. A class of models machine learning algorithms referred to as deep convolutional neural networks (CNNs) is revolutionising vegetation science (31) through its ability to exploit spatial structures and automatically extract high-level features from image data (e.g. analyses of satellite imagery (32)). In the field of computer vision, exactly segmenting individual objects of interest from an image is known as *instance segmentation*. The Mask R-CNN algorithm has shown promise in tree crown identification and delineation in plantations (33, 34), pine forests (35, 36), urban woodlands (35) and forest fragments (37). Here we use Facebook's pre-trained Mask R-CNN algorithm, (as *detectron2*) (38, 39), which harnesses the vast wealth of image data available to them. The *detectron2* framework has features that could allow it to overcome the challenges of delineating crowns in dense, complex tropical canopies by discriminating based on the spectral and textural signals which are rich due to the phylogenetic diversity.

In this paper, we describe Detectree2, an adaptation of Facebook AI's *Detectron2* system to detect tree crowns from aerial RGB imagery. We trained and evaluated Detectree2 on four tropical forest sites. In total, 2581 manually delineated tree crowns were used to train the model, and 1216 crowns were used to evaluate the model. We evaluated the performance with F_1 scores, which quantify the skill of the method in delineating individual tree crowns accurately. We expected to find that Detectree2 could be trained to delineate trees with high skill when trained with labels from a particular region but would perform poorly when tested on forest types different to those it was trained on, as neural architectures often struggle to generalise well. As an example of its ecological application, we deployed the trained model across 14 km^2 of airborne RGB imagery, automatically delineating 65,786 tree crowns. For context, this area is approximately half the total area of tropical forest inventory plots in the tropics. We then combined these predicted crowns with repeat airborne LiDAR data to investigate the growth and mortality rates of upper-canopy trees. Regional and global syntheses of forest inventory data suggest that growth slows down and mortality rates increase with tree size (40–44). We therefore expected to find the tallest trees we sampled to have lower growth rates and higher mortality rates than shorter trees. To explore this, we calculated the change in height for every tree and classed large negative changes as mortality events. We then tested whether growth or mortality events varied with initial tree height. The DetectTree2 package is available to install and apply on new regions.

Materials and Methods

Study sites. The analyses were conducted at four locations across three tropical field sites:

1. Sepilok Forest Reserve (East and West), Sabah, Malaysia (5°50'N, 177°56'W)
2. Danum Valley Conservation Area, Sabah, Malaysia (4°57'N, 177°41'W)
3. Paracou Field Station, French Guiana (5°16'N 52 °55'W)

The lowland tropical rain forests of Danum valley are dominated by dipterocarp species and are among the tallest forests on the planet (45). The available data from Sepilok included ecologically distinctive areas to the East and West. Sepilok West consists mostly of tall forest similar to Danum, while Sepilok East is a heath forest growing on shallow soils overlying sandstone, containing smaller trees that are more densely packed (18). All three sites in Malaysia experience a similar climate with approximately 2300 mm rainfall per year with the wettest months between November and February (46). Paracou contains lowland tropical rain forest mostly on shallow ferralitic soils underlain by a variably transformed loamy saprolite (47). The mean annual rainfall is approximately 3000 mm with a three month dry season from mid-August to mid-November (48). See Sup. Note 1 for more details on the study sites.

Remote sensing data. Airborne RGB surveys were conducted at all four sites using manned aircraft (Table 1). Repeat LiDAR surveys were also conducted at all four location (see Table 1, noting different sensors and altitudes between flights in Sabah). We analysed RGB imagery from 3.85 km^2 of Malaysian forest, with a ground resolution of 10 cm. In Paracou, we sampled 10.2 km^2 of imagery, with an 8-cm ground resolution. The raw imagery was orthorectified and collated into homogeneous mosaics using AgiSoft Metashape (49). The Structure from Motion method was used to calculate the elevation of the observed surface (50), and photographs were mapped onto this 3D surface. This process resulted in georeferenced photographs, largely corrected for distortions. Some distortion remained at the edges of the mosaic, so the analysis focused on the core areas where the imagery was not distorted.

Manual tree crown data. In order to train and then assess the accuracy of our automatic delineation approach, we created a manually labelled dataset of trees from all four sites

DRAFT

Table 1. Remote sensing data sources. Resolution is given as ground resolutions for the RGB imagery and returns per m^2 for LiDAR scans. The Sabah region includes both Sepilok and Danum sites.

Date	Region	Modality	Res.	Altitude	Sensor
10/2014	Sabah	RGB	10 cm	796 m	Leica RCD105
10/2014	Sabah	LiDAR	7.3 m^{-2}	2000 m	Leica ALS50
10/2020	Sabah	LiDAR	16 m^{-2}	250 m	Reigl LMS-Q780
19/09/2016	Paracou	RGB	8 cm	800 m	IXA180 Phase One
19/09/2016	Paracou	LiDAR	50 m^{-2}	800 m	RIEGL LMS-Q780
15/11/2019	Paracou	LiDAR	50 m^{-2}	800 m	RIEGL LMS-Q780

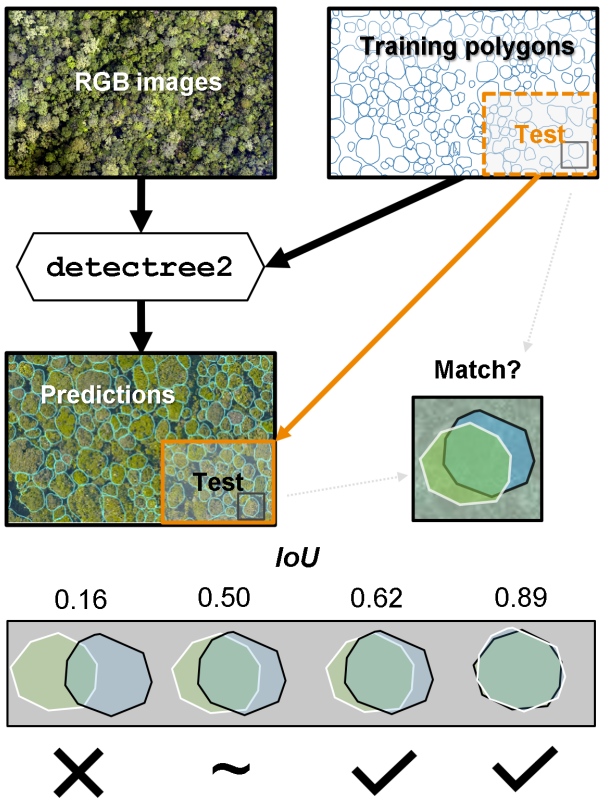

Fig. 1. The automatic tree crown delineation workflow. Manually delineated crowns are split into training and test sets. The Mask R-CNN framework combines the training set with the RGB imagery to learn how to delineate automatically from RGB images. A set of automatic predictions are produced across the entire RGB image and compared to the test set to evaluate the performance of the automatic delineations. IoU is used to determine when an automatic crown has been successfully matched with a manual crown.

using QGIS software (51). We generated our delineations using both RGB and LiDAR data, and in the case of Paracou supplementary hyperspectral layers. We used a number of techniques to improve the accuracy of crown delineation, including altering the contrast and saturation of the RGB image to exaggerate differences between the crowns, and using a mask of the LiDAR data to remove irrelevant parts of the RGB imagery. See Sup. Note 2 for further details.

We trained and tested our model on a total of 3797 manually delineated tree crowns across Paracou (training = 999, testing = 268), Danum (training = 371, testing = 150), Sepilok West (training = 628, testing = 410) and Sepilok East (training = 583, testing = 388). The crowns from Paracou were validated in the field with a expert local botanist, whereas the crowns in Malaysia were drawn purely by inspection of the remote sensing products. Four individuals performed the manual delineations which provided the network with variability in the inputs. This variability can be viewed as a data regularisation step which would likely improve the tranferability of the trained models. The trained model could therefore be seen as a synthesis of the human delineators, limited by their skill but harmonising their perspectives into a consistent output.

Both imagery and crown data required some processing before being fed into the network as described in Sup. Note 3.

Model architecture and parameterisation. Instance segmentation combines object detection with object segmentation. Once an object has been detected, a region of interest is established around the object. A segmentation is then carried out on the region of interest to identify the pixels within the region of interest which make up the object of interest (e.g. car, person, or tree crown) and which pixels are within the region of interest, but not of the object.

Here we use Facebook's Mask R-CNN algorithm, which was the best in class algorithm for instance segmentation when tested on the Microsoft COCO (Common Objects in Context) benchmark when initially released (38) and has since been updated (as *detectron2*) with improved training efficiency, documentation and transferability for integration into bespoke tools (39). We adapted Facebook AI's Detectron2 computer vision library to delineate individual tree crowns (39). The library is an implementation of the Mask R-CNN framework which is designed to identify and locate objects within images while also predicting object 'masks' which exactly circumscribe the objects in the image (i.e. *instance segmentation*, see Fig. S2, (38)). When initially re-

leased, Mask R-CNN was the state-of-the-art method for instance segmentation task on the Microsoft COCO benchmark (38, 52). The updated detectron2 implementation has improved training efficiency, adaptability, documentation and a 'model zoo'[1] from which specific model architectures with weights from a variety of pre-training regimes can be loaded. Taking a pre-trained model (weights) and retraining it to perform a novel task (e.g. delineating trees from aerial imagery) is an example of *transfer learning* (53) which can drastically reduce the amount of training data required to achieve acceptable performance on the new task.

We selected the R101-FPN configuration[2] as it has "the best speed/accuracy tradeoff" of the architectures available (39). The configuration sets the *backbone* of the network which is the part that views and extracts features from the scene as a whole. The R101-FPN backbone consists of a 101 layer deep ResNet (54) module with a Feature Pyramid Network (55) module. The initial model weights were generated from pre-training of the network on the ImageNet dataset[3]. Additional details are given in Sup. Note 5 but for full technical specifications, one should refer to the original papers and the detectron2 repository (38, 39).

Training procedure. Mask R-CNN is a supervised deep learning method and so requires labelled training examples as an input. Typically, a deep CNN would require several thousand training examples to learn a new task. This is a challenge in the case of tree crown delineation as manual delineation is very time consuming. The burden was reduced by transferring a pre-trained model. Additionally, the training data were augmented by applying several transformations (with random application through epochs) to the training cases including vertical and horizontal flips, rotation, and varying the saturation and contrast of the image. The model hyperparameters (see Table S1) were optimised with a Bayesian hyperparameter sweep implemented on wandb.ai[4]. This is an automated process that allows an automated agent to iteratively adjust hyperparameters to optimise based on a given accuracy metric. We selected the *AP50* of the segmentation predictions on the validation set. AP50 is the mean average precision when a correct match is granted when $IoU > 0.5$. See Sup. Note 4 for more details on model architecture, training and validation. The scripts on the GitHub repository[5] illustrate the best practices for using the model.

Computation. Model training and evaluation was performed on the Google Colab (Pro) platform which employed Intel(R) Xeon(R) CPU at 2.30GHz with 12.8 GB RAM and Tesla P100-PCIE-16GB GPUs. Model training always completed within 2 hours on this platform.

[1] https://github.com/facebookresearch/detectron2/blob/main/MODEL_ZOO.md
[2] https://github.com/facebookresearch/detectron2/blob/main/configs/COCO-Detection/faster_rcnn_R_101_FPN_3x.yaml
[3] https://www.image-net.org/
[4] https://wandb.ai/detectree/tune/sweeps/
[5] https://github.com/PatBall1/Detectree2

Performance evaluation. To determine the accuracy of delineation, we used the F_1 score metric, which compares the automatically predicted crown polygons with manually delineated tree crowns; the approach is not biased by tree crown area, and widely used in forest ecology (34, 37). The first metric required to calculate F_1 score is the area intersection over union (IoU).

$$IoU(A,B) = \frac{A \cap B}{A \cup B} \quad \textbf{(1)}$$

Where A is an automatically delineated crown, and B is a manually delineated crown. The F_1 Scores were determined by first calculating the IoU of all possible pairs of manually and automatically delineated trees. If the IoU of a particular pair was greater than 0.5, then the automatic delineation was considered to be a true positive. The threshold of 0.5 is widely used in similar studies (34). In rare cases where more than one automatic delineation had an IoU of greater than 0.5 with a manual crown, then the automatic delineation with the greater IoU was considered a true positive, the others being classed as false positives. The precision, recall and F_1 score were then determined. See Fig. 1 for an illustration of IoU and Sup. Note 5 for more details on the evaluation metrics.

Transferability across sites and resolutions. To further validate the performance and generality of Detectree2, we carried out some tests to evaluate the performance of the model when on different sites and resolutions of imagery. We first carried out a test where we trained the model on a single site and then tested it on our other sites, in order to determine whether our model was able to generalise across different areas of tropical forest. We then carried out another test, where we trained our model on the highest resolution available to us (10 cm), before testing it on lower resolution to determine the model's sensitivity to resolution. The final test was to train and test the model on low resolutions to determine if this workflow would increase the skill of the model on lower resolutions.

Application to monitoring growth and mortality. We combined our tree crown delineations from Detectree2 with repeat LiDAR surveys to determine the height changes in individual trees in our four sites. We determined the relationship between tree height and tree growth by fitting a robust least squares regression (56) to the data. Robust least squares was chosen to minimise the effects of outliers and mortality events on the regression. We note that hear we are measuring the vertical growth of trees which typically needs to be compared to inventory data through its allometric relationship to the diameter at breast height (DBH) growth.

To estimate mortality rates, we needed a suitable metric to identify mortality events. We took a statistical approach defining a mortality event as a negative change in height of more than three standard deviations below the robust least squares fit. This choice was ratified by manual inspection of trees meeting this threshold, and confirming that they constituted mortality events.

DRAFT

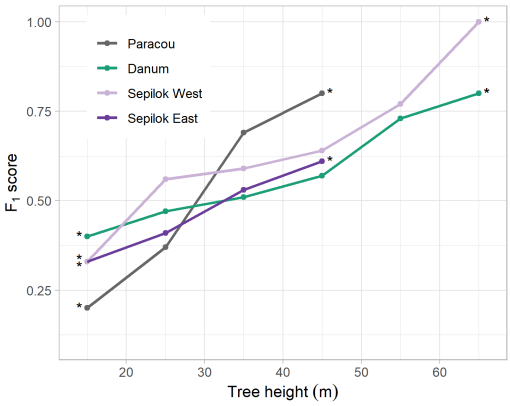

Fig. 2. Plot of the F_1 scores of Detectree2's when tested on trees taller than a certain height, across the four different sites. The points accompanied by an asterisk are those the number of trees in our testing set was fewer than 10.

We note that differences in LiDAR scanning parameters (pulse density, scanning angle etc.) can bias height estimates. Since this analysis is intended as a demonstration of the potential of Detectree2 in ecology and conservation we did not focus on this potential bias in the LiDAR data. This potential bias is more likely to affect our results on tree height growth (which are on the order of 0.1-0.5 m) rather than tree mortality (which are at least an order of magnitude larger).

Results

Validating automatic delineations. Detectree2 performed well ($F_1 > 0.6$) in Danum Valley, Sepilok West and Paracou. We are especially interested in the taller trees, for reasons outlined previously. To evaluate the performance of Detectree2 on tall trees, we isolated the trees above increasing height thresholds (e.g. we isolated all trees taller than 30m) in the manually and automatically delineated datasets, and then calculated the F_1 scores. The results of this evaluation are shown in Fig. 2, and show improved performance at greater heights. This is likely due to the increased size and visibility of tall trees in the RGB images, facilitating better delineation.

While the performance across the models varies by site, the trend of increased performance with height increase clear in all cases. Particularly for the site with the tallest trees, Sepilok West, we see excellent performance. It should also be noted that Sepilok East is a site with particularly small, dense trees and the model performs less well on this site.

Performance by site and resolution. Performance significantly degrades when testing on a different forest type to the one trained on. For example, the forest types in Sepilok East (mostly alluvial soils) and Sepilok West (heath) are different, and Detectree2's performance is significantly degraded when it is used to predict on Sepilok West when trained on data from Sepilok East, as illustrated in Figure 3.

To test the robustness of our model to imagery of varying resolutions we downscaled our RGB imagery to lower resolutions, and carried out model training and testing on these resolutions. Our resolutions were chosen as they are typical resolutions of satellite imagery.

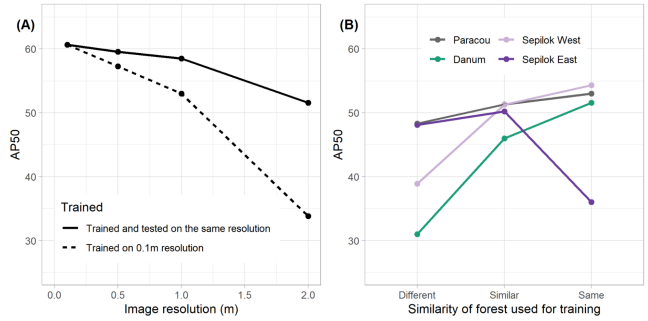

Fig. 3. Sensitivity of Detectree2 to (A) coarser image resolution and (B) similarity of the forest type used for training. Similarity is a subjective measure intended to describe the fact that both Sepilok West and Danum are tall dipterocarp dominated forests, whereas Sepilok East and Paracou are shorter forests with a larger number of trees per hectare. In addition, the image specifications were similar for all three Malaysian sites but quite different for Paracou (see methods)

Our results illustrate that our model trained on high resolution (0.1 m) imagery is most successful when predicting on similar resolutions, and its performance degrades for resolutions an order of magnitude greater (1 m and 2 m). However, we illustrate that when the model is trained on similar resolutions to those it is tested on, performance is largely maintained. It is only on resolutions greater than 1 m that the performance degrades significantly.

Application: Growth and mortality. One application of Detectree2 is to study tall tree growth and mortality rates. To do this, we overlaid Detectree2's tree crown predictions at the start date for each site on repeat LiDAR data (as canopy height models described in Sup. Note 1) to retrieve the tree height tree dynamics over time.

We were able to estimate the relationships between tree height and tree growth for each site by fitting robust least squares linear relationships between the two variables for Danum, Paracou, Sepilok East and Sepilok West. The regression coefficients and intercepts are given in Table S2.

The mortality rates for all four sites were calculated by taking the crowns predicted from the first collected RGB imagery, identifying the height of the delineated tree from the LiDAR data, and then calculating the change in height between the two years (2014 and 2020 for Sepilok and Danum, 2016 and 2019 for Paracou). The threshold for the required height change was taken by fitting a robust least squares regression to the height change against the original height of the tree, and taking trees that were three standard deviations below the mean to be mortality events. Figure 4 illustrates how certain trees were identified as mortality events, while some examples of mortality events from Paracou are given alongside.

Using this threshold for mortality events, we found overall annual mortality rates of 1.3% in both Sepilok West and Danum, 1.5% in Sepilok East and 1.9% in Paracou, as illustrated in Figure 5. The trees were then grouped into height bins of 10 metres (e.g. group all trees of height 30m to 40m). We then calculated the annual mortality rate in each height bin, as given in Figure 5. The mortality rates were highest for the tallest trees.

DRAFT

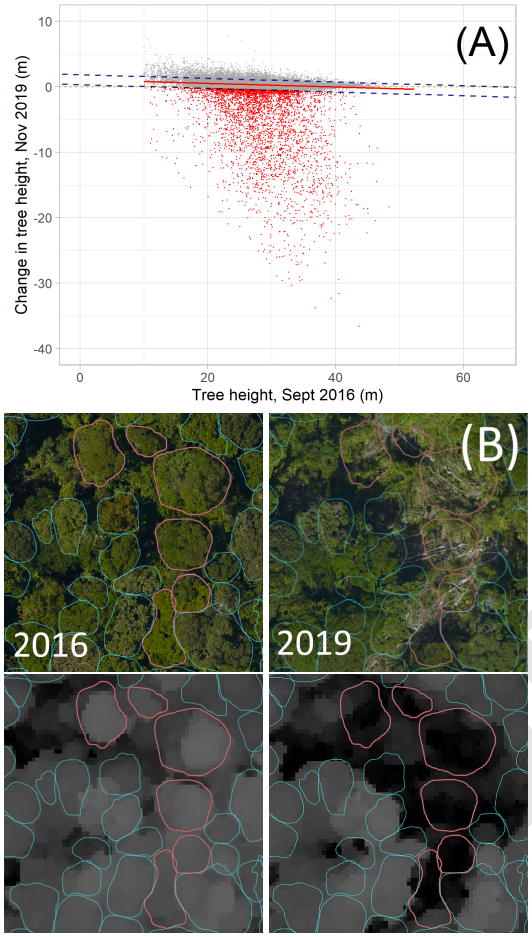

Fig. 4. Plot A shows the robust least squares fit and standard deviations. Plot B illustrates how predictions were overlaid on LiDAR data, and shows significant mortality events are outlined in red, and shown in the LiDAR and the RGB imagery. The 2016 imagery is shown on the left, 2019 on the right. Note that the crown delineations are based on the earlier imagery.

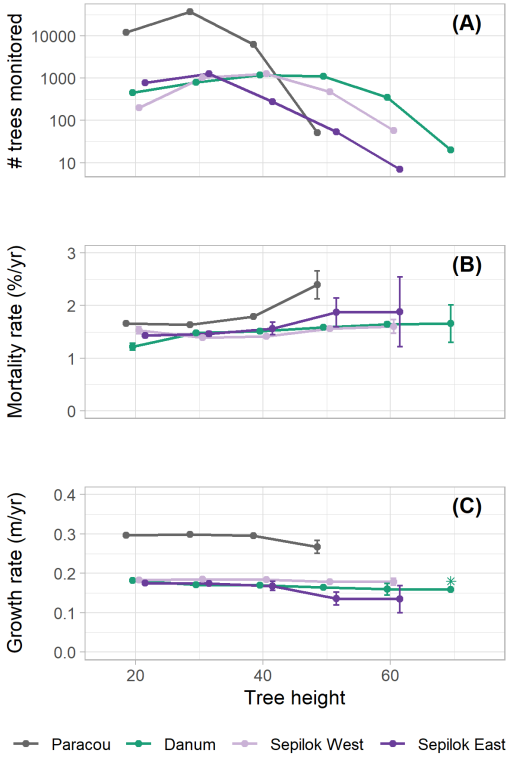

Fig. 5. Plot A shows the distribution of tree heights per site. Plot B shows the mortality rates of trees of different heights in each site, and Plot C gives the growth rate of trees split by height bin. Uncertainty estimates were determined by bootstrapping. The asterisk denotes an uncertainty estimate too large to fit on the plot.

Discussion

Improved tropical crown delineation. Accurately delineating trees in remote sensing data is a long-standing problem in ecology and conservation, and would enable us to efficiently monitor large areas of forests. Detectree2 addresses this problem, delineating individual trees in aerial RGB imagery with high precision and recall. We used Detectree2 to automatically delineate 65,786 trees across three tropical forests. We found that the accuracy of Detectree2 increased with tree height, meaning that that the tall trees which store the most carbon are also the most reliably delineated. We also show that Detectree2 generalises well across image resolutions (0.1 – 2 m pixel width), suggesting that it could be used with high resolution satellite data. If confirmed in practice, this could revolutionise forest remote sensing by enabling global scale individual tree detection in even the most complex forest types.

Detectree2 performed better than most existing methods, despite being tested in complex tropical forests. Furthermore, our model is able to delineate tree crowns exactly, rather than simply identifying the bounding box enclosing a tree. Detectree2 is also easier to use since it requires a relatively low number of manually delineated trees as training data compared to other methods (29, 37). These advantages are partly due to the fact that Detectree2 is based on a state-of-the-art pre-trained model, Mask R-CNN. Our results did not match the performance reported in (37). However, this study was based on 19,656 synthetic images of forests, and it is unclear

how the performance on real data would compare.

Generalisability across sites. We found that the accuracy was substantially lower when predicting on previously unseen forest or image types. However, we found that Detectree2 can be quickly trained to perform well on new areas of forest using around 10 images (each 1 ha in scale) with all visible tree crowns manually delineated. This manual delineation represents approximately 4 hours work.

The highest accuracy was achieved in Sepilok West and Paracou, for which we have a considerable number (10 ha and 14 ha respectively) of training images. The forest type in Sepilok West is also similar to that in Danum. The lowest accuracy was in Sepilok East, which had a smaller area (5 ha) of training images. We note that the manual delineation was done by different people focusing on separate sites. We were therefore unable to test for the effect of different delineators on our results as they are confounded with site differences. We note that this situation is representative of the expected applications where users with a particular study area will perform the manual delineation for that area. There may be opportunities to crowd-source training data and so compare different users' interpretations of the same image and their effects on the results, but this is beyond the scope of the current study.

Application: Growth and mortality rates. Tall trees store the majority of forest carbon and dominate many important forest nutrient cycles. However, they are rare and therefore poorly represented in traditional field inventories (44) which makes estimating their growth and mortality rates particularly challenging (40–43). Tall trees are also particularly sensitive to the effects of climate change (20), such as increased wind and storms, and as such tracking their dynamics over time is increasingly important. Recent remote sensing studies are bringing new insights into disturbance patterns by mapping the gaps left in the forest canopy after a tree (or multiple trees / branches) have fallen (57–59). Tracking individual trees over time instead of gaps will make it easier to interpret our results in an ecological context and also to compare the results more directly to the available field inventory data .

We demonstrate the value of Detectree2 in forest ecology by combining it with repeat LiDAR data to estimate growth and mortality rates of 65,786 trees over 14 km^2 of forest. This area is comparable to the total area of all field plots worldwide, which is estimated at 20 km^2 (ref). This illustrates the utility of our method for large scale mapping of tall trees.

We found substantially higher growth and mortality rates in French Guiana as compared to the sites in Malaysia. This suggests that these forests are more dynamic, with the increased growth being partly balanced by increased mortality. In addition, we found that taller trees had higher mortality rates and lower growth rates. This aligns with large scale analyses of field-based studies (40) and demonstrates the usefulness of tracking tall trees in this way.

We note three important caveats in our analysis of growth and mortality rates. Firstly, they are based purely on height data derived from LiDAR, so the growth reported is vertical growth and therefore not directly comparable to that measured in field inventories. Secondly, we defined mortality as a drop in height of more than a statistically determined threshold. We cannot therefore verify that the tree has died, although it is likely that it has snapped or uprooted. Finally, the fact that our predictions are less accurate for shorter trees may bias the observed trends in vertical tree growth and mortality. Nevertheless, we believe this example application demonstrates the utility of Detectree2 in forest ecology.

Future methodological developments and applications. As shown in Figure 2, Detectree2 performs impressively when delineating tall trees. However, it fails to delineate a significant proportion of trees. There is considerable scope to increase the quantity and variety of training data for Mask R-CNN by labelling more trees by hand, as evidenced by recent work using Mask R-CNN for a similar purpose (37).

Furthermore, Mask R-CNN suffers from problems typical in CNN-based architectures. Firstly, the problem of scale - if trained on images where trees are a certain size, it struggles to predict on images where trees are relatively smaller or larger, as a proportion of the whole image. Secondly, if given training data that includes images that are not completely labelled, the model does not learn as effectively since it learns that certain trees are not in fact trees, as they have not been labelled. It would be more practical if the model could be modified to train only in a positive sense, since manually labelling all the trees in a tile (~4 ha) of forest is difficult and time intensive.

The fact that Detectree2 can be quickly trained to perform well on a new type of forest and imagery demonstrates that it is a useful tool for forest management. Many conservation or restoration projects have access to low-cost imagery from drones or satellites. Detectree2 would allow them to quickly quantify and track the number and size distribution of trees across an entire landscape. In combination with other remote sensing data sources, this could allow for improved carbon stock and dynamics estimation. Estimating carbon stocks in forests has traditionally been done using area-based methods which discard considerable granular information at the individual tree level (18).

While we studied the delineation of a single class (tree), Detectree2 is capable of being trained and making predictions on multiple classes. This may allow for low cost species identification and mapping. It also presents the possibility of automatically assessing liana infestation occurrence . Previously, hyperspectral data has been employed to address this problem but with limited success due to the phylogenetic and spectral diversity of lianas (60, 61).

Our tests suggest that Detectree2 can accurately delineate trees in RGB imagery at 2 m resolution. This means that it should be possible to delineate trees in high resolution satellite imagery, and therefore to obtain global coverage. If this proves possible it will help answer many long-standing questions in forest ecology as well as providing an important tool for forest management.

DRAFT

Conclusions

In this study we introduced and tested Detectree2, a method for delineating individual tree crowns using a deep learning method. Our model shows considerable skill, particularly in identifying the tallest trees in the canopy.

We combined Detectree2 with repeat airborne LiDAR data for sites in Malaysia and French Guiana to extract information tree growth and mortality. This enabled us to track around 65,786 trees over an area of 14 km^2. We found that tree growth decreases and mortality increases with increasing tree height, which aligns with previous field studies.

Initial tests suggest that Detectree2 can delineate trees in high resolution satellite data, which would enable global scale monitoring of forests at the individual tree level.

ACKNOWLEDGEMENTS

We thank the NERC Earth Observation Data Acquisition and Analysis Service (NEODAAS) for supplying data and computational resources for this study. We thank CEDA for the 2014 Sepilok and Danum remote sensing data. SH received funding from the Centre for Doctoral Training in Application of Artificial Intelligence to the study of Environmental Risks (AI4ER, EP/S022961/1), which is supported by the Engineering and Physical Sciences Research Council (EPSRC). JGCB was supported by the NERC C-CLEAR doctoral training programme. TDJ and DAC were supported by NERC grant [NE/S010750/1]. DAC was supported by the Franklinia Foundation. We thank Stephen Goult for contributing to discussions. Data collection in French Guiana was supported by CNES who funded the 2016 hyperspectral, RGB and Lidar data over Paracou and Labex CEBA (ANR-10-LABX-25) for contributing financial resource for the field validation of manual crown segmentations. The 2019 data in Paracou and 2020 data in Sabah were funded by NERC [NE/S010750/1]. The 2014 Sabah data were also funded by NERC [NE/K016377/1].

Bibliography

1. Yude Pan, Richard A. Birdsey, Jingyun Fang, et al. A large and persistent carbon sink in the world's forests. *Science*, 333(6045):988–993, 2011. doi: 10.1126/science.1201609.
2. R J W Brienen, O L Phillips, T R Feldpausch, et al. Long-term decline of the amazon carbon sink. *Nature*, 519:344–348, 2015. ISSN 1476-4687. doi: 10.1038/nature14283.
3. Wannes Hubau, Simon L Lewis, Oliver L Phillips, et al. Asynchronous carbon sink saturation in african and amazonian tropical forests. *Nature*, 579:80–87, 2020. ISSN 1476-4687. doi: 10.1038/s41586-020-2035-0.
4. Yuanfang Chai, Guilherme Martins, Carlos Nobre, et al. Constraining amazonian land surface temperature sensitivity to precipitation and the probability of forest dieback. *npj Climate and Atmospheric Science*, 4:6, 2021. ISSN 2397-3722. doi: 10.1038/s41612-021-00162-1.
5. Rosie A. Fisher, Charles D. Koven, William R. L. Anderegg, et al. Vegetation demographics in earth system models: A review of progress and priorities. *Global Change Biology*, 24(1): 35–54, 2018. doi: https://doi.org/10.1111/gcb.13910.
6. Charles D Koven, Ryan G Knox, Rosie A Fisher, et al. Benchmarking and parameter sensitivity of physiological and vegetation dynamics using the functionally assembled terrestrial ecosystem simulator (fates) at barro colorado island, panama. *Biogeosciences*, 17(11): 3017–3044, 2020.
7. Sarah M. Gallup, Ian T. Baker, John L. Gallup, et al. Accurate simulation of both sensitivity and variability for amazonian photosynthesis: Is it too much to ask? *Journal of Advances in Modeling Earth Systems*, 13(8):e2021MS002555, 2021. doi: https://doi.org/10.1029/2021MS002555. e2021MS002555 2021MS002555.
8. Natalia Restrepo-Coupe, Loren P. Albert, Marcos Longo, et al. Understanding water and energy fluxes in the amazonia: Lessons from an observation-model intercomparison. *Global Change Biology*, 27(9):1802–1819, 2021. doi: https://doi.org/10.1111/gcb.15555.
9. Camille Piponiot, Kristina J. Anderson-Teixeira, Stuart J. Davies, et al. Distribution of biomass dynamics in relation to tree size in forests across the world. *New Phytologist*, 234(5):1664–1677, 2022. doi: https://doi.org/10.1111/nph.17995.
10. Pieter A. Zuidema and Peter van der Sleen. Seeing the forest through the trees: how tree-level measurements can help understand forest dynamics. *New Phytologist*, 234(5): 1544–1546, 2022. doi: https://doi.org/10.1111/nph.18144.
11. Michele Dalponte and David A. Coomes. Tree-centric mapping of forest carbon density from airborne laser scanning and hyperspectral data. *Methods in Ecology and Evolution*, 7(10): 1236–1245, 2016. doi: https://doi.org/10.1111/2041-210X.12575.
12. Jérôme Chave, Stuart J Davies, Oliver L Phillips, et al. Ground data are essential for biomass remote sensing missions. *Surveys in Geophysics*, 40:863–880, 2019. ISSN 1573-0956. doi: 10.1007/s10712-019-09528-w.
13. Stuart J. Davies, Iveren Abiem, Kamariah Abu Salim, et al. Forestgeo: Understanding forest diversity and dynamics through a global observatory network. *Biological Conservation*, 253: 108907, 2021. ISSN 0006-3207. doi: https://doi.org/10.1016/j.biocon.2020.108907.
14. ForestPlots.net, Cecilia Blundo, Julieta Carilla, et al. Taking the pulse of earth's tropical forests using networks of highly distributed plots. *Biological Conservation*, 260:108849, 2021. ISSN 0006-3207. doi: https://doi.org/10.1016/j.biocon.2020.108849.
15. David C. Marvin, Gregory P. Asner, David E. Knapp, et al. Amazonian landscapes and the bias in field studies of forest structure and biomass. *Proceedings of the National Academy of Sciences*, 111(48):E5224–E5232, 2014. doi: 10.1073/pnas.1412999111.
16. Victoria Meakem, Alan J. Tepley, Erika B. Gonzalez-Akre, et al. Role of tree size in moist tropical forest carbon cycling and water deficit responses. *New Phytologist*, 219(3):947–958, 2018. doi: https://doi.org/10.1111/nph.14633.
17. James A Lutz, Tucker J Furniss, Daniel J Johnson, et al. Global importance of large-diameter trees. *Global Ecology and Biogeography*, 27(7):849–864, 2018.
18. David A Coomes, Michele Dalponte, Tommaso Jucker, et al. Area-based vs tree-centric approaches to mapping forest carbon in southeast asian forests from airborne laser scanning data. *Remote Sensing of Environment*, 194:77–88, 2017.
19. Atticus E L Stovall, Herman Shugart, and Xi Yang. Tree height explains mortality risk during an intense drought. *Nature Communications*, 10:4385, 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-12380-6.
20. Evan M Gora and Adriane Esquivel-Muelbert. Implications of size-dependent tree mortality for tropical forest carbon dynamics. *Nature Plants*, 7:384–391, 2021. ISSN 2055-0278. doi: 10.1038/s41477-021-00879-0.
21. IPCC. *Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press, 2021.
22. Zhen Zhen, Lindi J. Quackenbush, and Lianjun Zhang. Trends in automatic individual tree crown detection and delineation—evolution of lidar data. *Remote Sensing*, 8(4), 2016. ISSN 2072-4292. doi: 10.3390/rs8040333.
23. Jack H. Hastings, Scott V. Ollinger, Andrew P. Ouimette, et al. Tree species traits determine the success of lidar-based crown mapping in a mixed temperate forest. *Remote Sensing*, 12(2), 2020. ISSN 2072-4292. doi: 10.3390/rs12020309.
24. Mélaine Aubry-Kientz, Raphaël Dutrieux, Antonio Ferraz, et al. A comparative assessment of the performance of individual tree crowns delineation algorithms from als data in tropical forests. *Remote Sensing*, 11(9), 2019. ISSN 2072-4292. doi: 10.3390/rs11091086.
25. Mélaine Aubry-Kientz, Anthony Laybros, Ben Weinstein, et al. Multisensor data fusion for improved segmentation of individual tree crowns in dense tropical forests. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:3927–3936, 2021. doi: 10.1109/JSTARS.2021.3069159.
26. Jakob Iglhaut, Carlos Cabo, Stefano Puliti, et al. Structure from motion photogrammetry in forestry: a review. *Current Forestry Reports*, 5:155–168, 2019. ISSN 2198-6436. doi: 10.1007/s40725-019-00094-3.
27. André Almeida, Fabio Gonçalves, Gilson Silva, et al. Individual tree detection and qualitative inventory of a eucalyptus sp. stand using uav photogrammetry data. *Remote Sensing*, 13 (18), 2021. ISSN 2072-4292. doi: 10.3390/rs13183655.
28. Anderson Aparecido dos Santos, Jose Marcato Junior, Márcio Santos Araújo, et al. Assessment of cnn-based methods for individual tree detection on images captured by rgb cameras attached to uavs. *Sensors*, 19(16):3595, 2019.
29. Ben G Weinstein, Sergio Marconi, Stephanie Bohlman, et al. Individual tree-crown detection in rgb imagery using semi-supervised deep learning neural networks. *Remote Sensing*, 11 (11):1309, 2019.
30. Ben G Weinstein, Sergio Marconi, Stephanie A Bohlman, et al. A remote sensing derived data set of 100 million individual tree crowns for the national ecological observatory network. *Elife*, 10:e62922, 2021.
31. Teja Kattenborn, Jens Leitloff, Felix Schiefer, and Stefan Hinz. Review on convolutional neural networks (cnn) in vegetation remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 173:24–49, 2021. ISSN 0924-2716. doi: https://doi.org/10.1016/j.isprsjprs.2020.12.010.
32. Xiao Xiang Zhu, Devis Tuia, Lichao Mou, et al. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8–36, 2017. doi: 10.1109/MGRS.2017.2762307.
33. Kunyong Yu, Zhenbang Hao, Christopher J. Post, et al. Comparison of classical methods and mask r-cnn for automatic tree detection and mapping using uav imagery. *Remote Sensing*, 14(2), 2022. ISSN 2072-4292. doi: 10.3390/rs14020295.
34. Zhenbang Hao, Lili Lin, Christopher J Post, et al. Automated tree-crown and height detection in a young forest plantation using mask region-based convolutional neural network (mask r-cnn). *ISPRS Journal of Photogrammetry and Remote Sensing*, 178:112–123, 2021.
35. Nuri Erkin Ocer, Gordana Kaplan, Firat Erdem, et al. Tree extraction from multi-scale uav images using mask r-cnn with fpn. *Remote Sensing Letters*, 11(9):847–856, 2020. doi: 10.1080/2150704X.2020.1784491.
36. Gensheng Hu, Tongxiang Wang, Mingzhu Wan, et al. Uav remote sensing monitoring of pine forest diseases based on improved mask r-cnn. *International Journal of Remote Sensing*, 43(4):1274–1305, 2022. doi: 10.1080/01431161.2022.2032455.
37. José R G Braga, Vinícius Peripato, Ricardo Dalagnol, et al. Tree crown delineation algorithm based on a convolutional neural network. *Remote Sensing*, 12(8):1288, 2020.
38. Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.
39. Yuxin Wu, Alexander Kirillov, Francisco Massa, et al. Detectron2. https://github.com/facebookresearch/detectron2, 2019.
40. Yoshiko Iida, Lourens Poorter, Frank Sterck, et al. Linking size-dependent growth and mortality with architectural traits across 145 co-occurring tropical tree species. *Ecology*, 95(2): 353–363, 2014.
41. Sarah J Richardson, Mark C Smale, Jennifer M Hurst, et al. Large-tree growth and mortality rates in forests of the central north island, new zealand. *New Zealand Journal of Ecology*, pages 208–215, 2009.
42. David A Coomes, Richard P Duncan, Robert B Allen, and James Truscott. Disturbances prevent stem size-density distributions in natural forests from following scaling relationships. *Ecology letters*, 6(11):980–989, 2003.
43. Helene C Muller-Landau, Richard S Condit, Jerome Chave, et al. Testing metabolic ecology theory for allometric scaling of tree size, growth and mortality in tropical forests. *Ecology letters*, 9(5):575–588, 2006.
44. Jennifer M Hurst, Robert B Allen, David A Coomes, and Richard P Duncan. Size-specific

DRAFT

tree mortality varies with neighbourhood crowding and disturbance in a montane nothofagus forest. *PloS one*, 6(10):e26670, 2011.
45. Alexander Shenkin, Chris J Chandler, Doreen S Boyd, et al. The world's tallest tropical tree in three dimensions. *Frontiers in Forests and Global Change*, 2:32, 2019.
46. Reuben Nilus, Colin R Maycock, N Majalap-Lee, and DFRP Burslem. Nutrient limitation of tree seedling growth in three soil types found in sabah. *Journal of Tropical Forest Science*, pages 133–142, 2011.
47. Sylvie Gourlet-Fleury, Jean-Marc Guehl, and Olivier Laroussinie. *Ecology and management of a neotropical rainforest: lessons drawn from Paracou, a long-term experimental research site in French Guiana*. Elsevier, 2004.
48. Fabien Wagner, Bruno Hérault, Clément Stahl, et al. Modeling water availability for trees in tropical forests. *Agricultural and Forest Meteorology*, 151(9):1202–1213, 2011. ISSN 0168-1923. doi: https://doi.org/10.1016/j.agrformet.2011.04.012.
49. AgiSoft. Agisoft metashape (version 1.7.3), 2021.
50. Matthew J Westoby, James Brasington, Niel F Glasser, et al. 'structure-from-motion'photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, 179:300–314, 2012.
51. QGIS Association. Qgis geographic information system, 2021.
52. Tsung-Yi Lin, Michael Maire, Serge Belongie, et al. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
53. Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016.
54. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
55. Tsung-Yi Lin, Piotr Dollár, Ross Girshick, et al. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.
56. Jean-Yves Audibert and Olivier Catoni. Robust linear least squares regression. *The Annals of Statistics*, 39(5):2766–2794, 2011.
57. Raquel Fernandes Araujo, Samuel Grubinger, Carlos Henrique Souza Celes, et al. Strong temporal variation in treefall and branchfall rates in a tropical forest is related to extreme rainfall: results from 5 years of monthly drone data for a 50 ha plot. *Biogeosciences*, 18(24): 6517–6531, 2021.
58. KC Cushman, Matteo Detto, Milton García, and Helene C Muller-Landau. Soils and topography control natural disturbance rates and thereby forest structure in a lowland tropical landscape. *Ecology Letters*, 2022.
59. Claudia Huertas, Daniel Sabatier, Géraldine Derroire, et al. Mapping tree mortality rate in a tropical moist forest using multi-temporal lidar. *International Journal of Applied Earth Observation and Geoinformation*, 109:102780, 2022. ISSN 1569-8432. doi: https://doi.org/10.1016/j.jag.2022.102780.
60. Mathias Wessel, Melanie Brandmeier, and Dirk Tiede. Evaluation of different machine learning algorithms for scalable classification of tree types and tree species based on sentinel-2 data. *Remote Sensing*, 10(9):1419, 2018.
61. Ewa Grabska, David Frantz, and Katarzyna Ostapowicz. Evaluation of machine learning algorithms for forest stand species mapping using sentinel-2 imagery and environmental data in the polish carpathians. *Remote Sensing of Environment*, 251:112103, 2020.
62. Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.
63. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.
64. Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

DRAFT

Word Counts

This section is *not* included in the word count.

Notes on Methods in Ecology and Evolution.

- Abstract: 350 words and should list the main results and conclusions, using FOUR simple, factual, numbered statements
- Main text: no more than 7000-8000 words for Standard Articles, 3000-4000 words for Applications and Practical Tools

Statistics on word count.

```
!!! File Article.bbl not found in path []. !!!
File: Article.tex
Encoding: utf8
Sum count: 5667
Words in text: 5200
Words in headers: 80
Words outside text (captions, etc.): 378
Number of headers: 24
Number of floats/tables/figures: 6
Number of math inlines: 8
Number of math displayed: 1
Subcounts:
  text+headers+captions (#headers/#floats/#inlines/#displayed)
  94+13+0 (1/0/0/0) _top_
  326+1+0 (1/0/0/0) Abstract
  26+0+0 (0/0/0/0) main
  888+1+0 (1/0/0/0) Section: Introduction
  0+3+0 (1/0/0/0) Section: Materials and Methods
  188+2+0 (1/0/0/0) Subsection: Study sites
  137+3+31 (1/1/0/0) Subsection: Remote sensing data
  250+4+79 (1/1/1/0) Subsection: Manual tree crown data
  359+4+24 (1/0/0/0) Subsection: Model architecture and parameterisation
  186+2+9 (1/0/1/0) Subsection: Training procedure
  39+1+0 (1/0/0/0) Subsection: Computation
  187+2+0 (1/0/6/1) Subsection: Performance evaluation
  131+5+0 (1/0/0/0) Subsection: Transferability across sites and resolutions
  234+6+0 (1/0/0/0) Subsection: Application to monitoring growth and mortality
  0+1+0 (1/0/0/0) Section: Results
  163+3+43 (1/1/0/0) Subsection: Validating automatic delineations
  174+5+74 (1/1/0/0) Subsection: Performance by site and resolution
  292+4+118 (1/2/0/0) Subsection: Application: Growth and mortality
  0+1+0 (1/0/0/0) Section: Discussion
  243+4+0 (1/0/0/0) Subsection: Improved tropical crown delineation
  211+3+0 (1/0/0/0) Subsection: Generalisability across sites
  381+5+0 (1/0/0/0) Subsection: Application: Growth and mortality rates
  403+5+0 (1/0/0/0) Subsection: Future methodological developments and applications
  287+1+0 (1/0/0/0) Section: Conclusions
  1+1+0 (1/0/0/0) Section: Bibliography

(errors:1)
```

DRAFT

Supplementary Note 1: Study sites and remote sensing data collection

A. Sepilok and Danum. Sepilok Forest Reserve (5°50' N 117°56' E) is a region of lowland tropical rainforest in Sabah, a state in Malaysia. The Reserve is one of the oldest protected areas of forest in Asia, being founded by the Sabah Forest Department in 1931. The Reserve spans nearly 4500 ha, with ground elevation varying between 50 and 250 metres above sea level. Three distinct forest types are present: alluvial dipterocarp, sandstone dipterocarp and heath forest. Danum Forest is... Our 2014 LiDAR data of Sepilok and Danum was collected using a Leica ALS50-II ALS flown at an altitude of nearly 2000 metres, attached to the belly of a Dornier 228-201. The ALS sensor emitted pulses at around 80 Hz with a field of view of 12.0°, and a footprint of about 40 cm diameter. The average pulse density was 7.3 m-2. The ALS data were preprocessed by NERC's Data Analysis Node and delivered in LAS format. The 2020 LiDAR data in Sepilok was collected using a Riegl LMS-Q780 mounted on a helicopter flying at 250 metres altitude at a ground speed of approximately 100 km/hr. This resulted in an average pulse density of 16 m-2. Further processing used LAStools (http://rapidlasso.com/lastools/). Points were split into two groups, ground and non-ground, and a digital elevation model (DEM) was fitted to the ground points, producing a raster of 0.5 m resolution. The DEM elevations were subtracted from elevations of all non-ground returns, known as the digital surface model (DSM) to create a canopy height model (CHM) raster of resolution 0.5 m.

The 2014 RGB imagery was collected using a Leica RCD105 Digital Camera, attached to a plane flown at an altitude of 796 metres, with a ground resolution of 10 cm. Photographs have been orthorectified and collated into homogenous mosaics using Agisoft Metashape. This software used the Structure from Motion method to calculate the elevation of the observed surface, of which the photographs are mapped to, This results in georeferenced photographs that are corrected for distortions. The files were delivered in the GeoTIFF format.

B. Paracou. The Paracou field station is situated in a lowland tropical forest near Sinnamary, French Guiana (5°16'N 52 °55'W). It is a forest similar to Sepilok Forest Reserve, which is also a lowland tropical forest.

ALS data were acquired in September 2016, and November 2019 by ALTOA, operating a RIEGL LMS-Q780 sensor attached to an aircraft flying at 800m. On all dates, the scan frequency was 400 kHz and the final point density was above 50 m^{-2}. The creation of a CHM was carried out using the same process as used for Sepilok.

On the same flights as the ALS scans, RGB images were collected using an IXA180 Phase One camera with an 8 cm ground sampling distance.

Hyperspectral imagery was also collected over Paracou, with a Hyspex VNIR-1600 (Hyspex NEO, Skedsmokorset, Norway) sensor-mounted alongside the RIEGL scanner. Its bands covered a spectral range of 414–994 nm with a spectral sampling distance of 3.64 nm. Images were orthorectified and georeferenced to 1 m spatial resolution with the PARGE software using the DSM from the LiDAR data.

DRAFT

Supplementary Note 2: Tree crown data

A. Manual delineation. Manual tree crown labelling was carried out in two areas of tropical forest, Sepilok Forest Reserve and Paracou (French Guiana), to create training and test sets for our analysis. Labelling was carried out using a combination of RGB, LiDAR, and in the case of Paracou, hyperspectral imagery, all loaded into QGIS (51), software that allows visualisation of geospatial data. To distinguish individual trees more easily, the colours of the RGB imagery were stretched, as detailed in Table S3.1. A shapefile layer was then created in QGIS, and polygons were carefully drawn around the perimeter of all distinguishable trees for a number of areas in each region. JB carried out the manual labelling for Paracou.

A visualisation of the difference between the original and the stretched image is given in Figure S1.

B. Training and validation data. The algorithm was trained on 999 examples of manually delineated tree crowns, and validated with 268 manually delineated crowns from Paracou in French Guiana. Examples of the training data are given in Figure S4.

Supplementary Note 3: Data pre-processing

To apply *Detectree2* to remote sensing RGB images of tropical forests, the geospatial raster image must be tiled and converted into a png format, and the manual crown segmentations used for training must also be converted from geospatial shapefiles into JSON files that align with the pngs. These training tiles are ingested into Detectree2, which then learns how to identify a tree, and then can make predictions on new tiles.

Supplementary Note 4: Detectron2 library and Mask R-CNN framework

A. Model architecture. Mask R-CNN (38) is a framework for instance segmentation. It builds on Faster R-CNN (62), which carries out object detection using a Region Proposal Network (RPN). An RPN is a fully convolutional network, trained end-to-end, that generates Regions of Interest (RoIs) for each image. These RoIs have object bounds and objectness scores attached

Fig. S1. An original RGB image is displayed on the left, while a stretched RGB image is displayed on the right of this figure. Stretching the colours of the image to the values given in Table S3.1 allows for easier identification of individual trees when carrying out manual tree crown delineations. The effect is particularly noticeable in the lower left corner of the images.

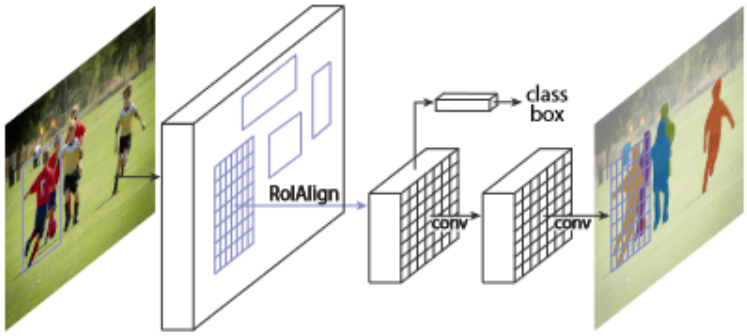

Fig. S2. Illustration of the Mask R-CNN architecture from He et al. (38).

to them, giving the bounds of the RoI, and the likelihood of the RoI containing an object. These RoIs are then passed through fully connected convolutional networks to determine the class of image contained, and the exact mask of each object, within the respective bounding box. For full details on the structure of Mask R-CNN, please refer to the GitHub repository for this work (https://github.com/PatBall1/Detectree2). The schematic of the architecture of the model architecture is given in Figure S2.

We selected the R101-FPN configuration - this architecture combines a 101 layer deep ResNet (63) module with a Feature Pyramid Network module.

This model is an improvement upon existing architectures, such as architectures performing object detection. An example of the different predictions is given in Figure S3.

B. Data augmentation. The training data was augmented by submitting the training data to a variety of transformations including vertical and horizontal flips, rotation, and varying the saturation and contrast of the images. These augmentations are designed to increase the variety of training data seen by the model, to allow it to generalise better to new images.

C. Hyperparameter tuning. As with all deep networks, there were a number of hyperparameters that control the training of the algorithm. The first of these relates to the architecture of the algorithm, namely the number of hidden layers in the ResNet backbone (54) of the network. ResNets are used as the backbone of Mask R-CNN as very deep neural networks are particularly difficult to train, due to the vanishing gradients problem (64). ResNets use skip connections, whereby certain layers in the network are skipped during backpropagation to avoid this problem.

Other hyperparameters that can be optimised in Mask R-CNN relate to the training of the network. Namely the learning rate, the number of iterations and the batch size. The optimisation of these hyperparameters was carried out using Weights and Biases, which uses grid search to determine the optimal hyperparameters. The hyperparameters selected are given in Table

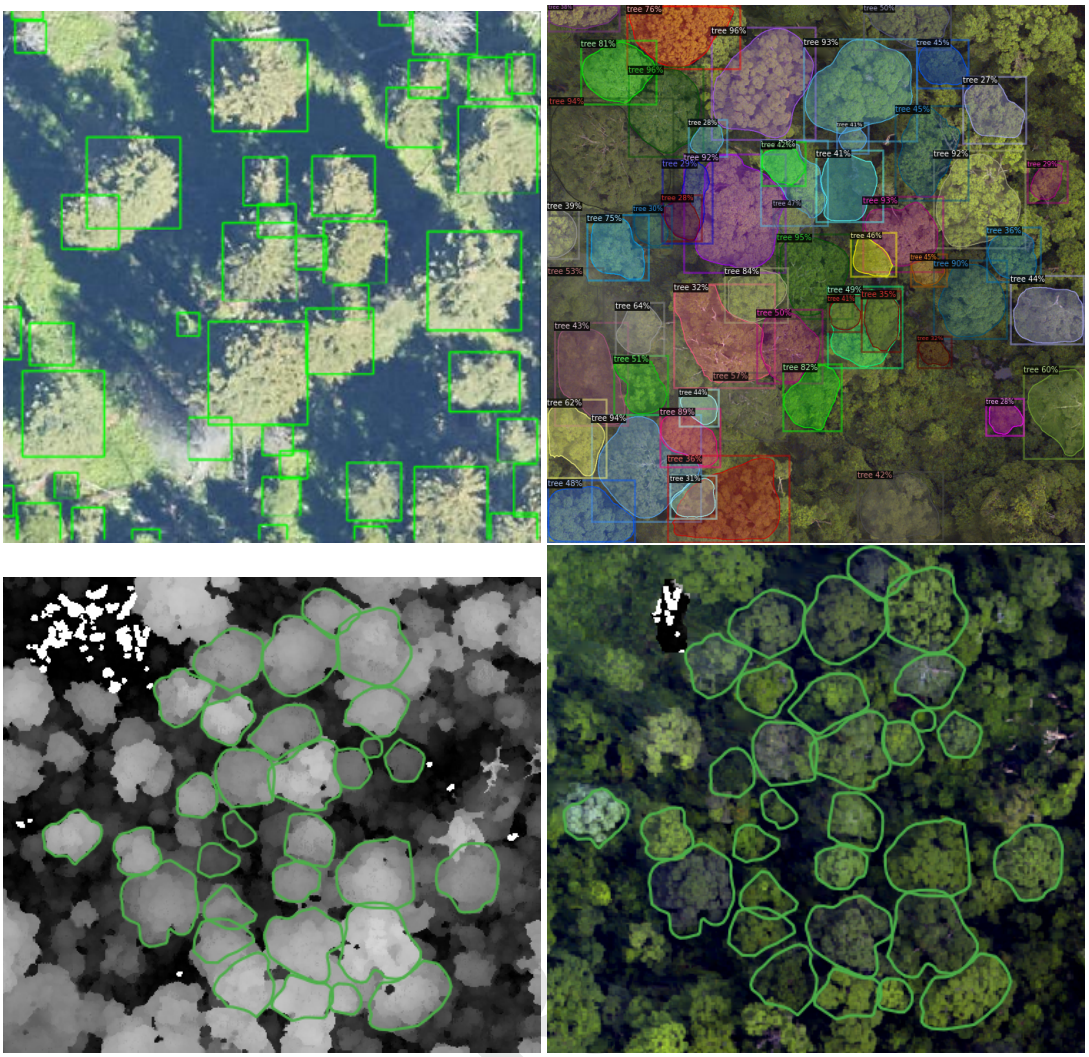

Fig. S3. Crowns predicted by DeepForest (Weinstein et al., 2019) (A), and crowns predicted by Detectree2 (B). The colours in plot B merely distinguish predicted trees. A comparison of manually delineated crowns, overlaid on LiDAR (C) and RGB (D).

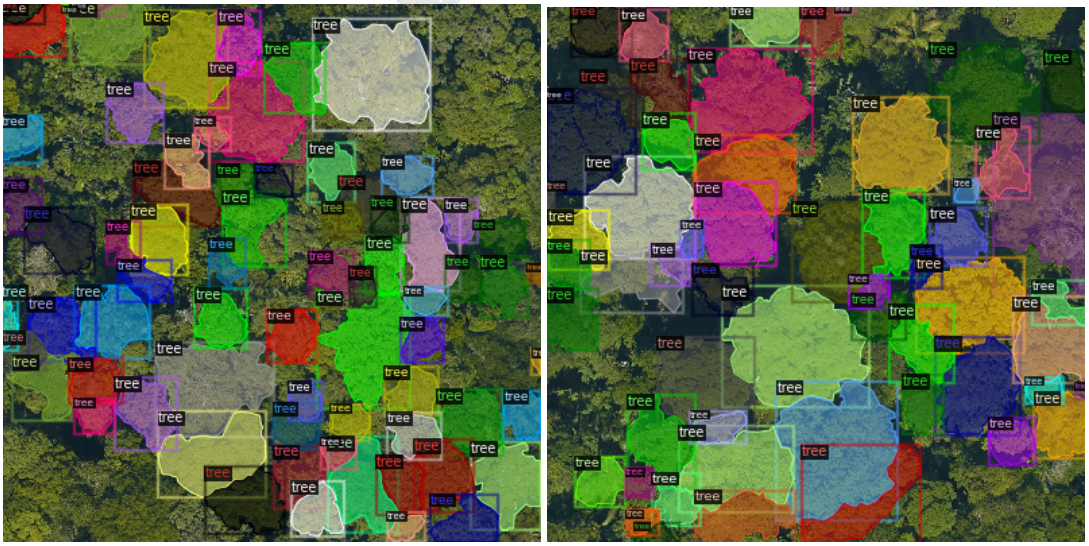

Fig. S4. Examples of training data provided to Mask R-CNN. The different colours help to distinguish between trees.

Table S1. Tunable hyperparameters (with their optimised value) and a description of their purpose.

Hyperparameter (=val)	Description
dl_nums_workers (=2)	To speed up the training process, we make use of the num_workers optional attribute of the DataLoader class. The num_workers attribute tells the data loader instance how many sub-processes to use for data loading. By default, the num_workers value is set to zero, and a value of zero tells the loader to load the data inside the main process. This means that the training process will work sequentially inside the main process. After a batch is used during the training process and another one is needed, we read the batch data from the disk. Now, if we have a worker process, we can make use of the fact that our machine has multiple cores. This means that the next batch can already be loaded and ready to go by the time the main process is ready for another batch. This is where the speed up comes from. The batches are loaded using additional worker processes and are queued up in memory.
ims_per_batch (=2)	If we use 16 GPUs and IMS_PER_BATCH = 32, each GPU will see 2 images per batch.
gamma (=0.1)	The iteration number to decrease the learning rate by GAMMA.
backbone_freeze_at (=3)	Freeze the first several stages so they are not trained. There are 5 stages in ResNet. The first is a convolution, and the following stages are each group of residual blocks. Freezing a layer prevents its weights from being modified. This technique is often used in transfer learning, where the base model(trained on some other dataset is frozen.
warmup_iters (=120)	Warmup steps are just a few updates with low learning rate before training. After this warmup, you use the regular learning rate, (schedule) to train your model to convergence. The idea that this helps your network to slowly adapt to the data intuitively makes sense. If your data set is highly differentiated, you can suffer from a sort of "early over-fitting". If your shuffled data happens to include a cluster of related, strongly-featured observations, your model's initial training can skew badly toward those features - or worse, toward incidental features that aren't truly related to the topic at all. Warm-up is a way to reduce the primacy effect of the early training examples. Without it, you may need to run a few extra epochs to get the convergence desired, as the model un-trains those early superstitions. Many models afford this as a command-line option. The learning rate is increased linearly over the warm-up period. If the target learning rate is p and the warm-up period is n, then the first batch iteration uses $1p/n$ for its learning rate; the second uses $2p/n$, and so on: iteration i uses $i * p/n$, until we hit the nominal rate at iteration n. This means that the first iteration gets only $1/n$ of the primacy effect. This does a reasonable job of balancing that influence. Note that the ramp-up is commonly on the order of one epoch – but is occasionally longer for particularly skewed data, or shorter for more homogeneous distributions. You may want to adjust, depending on how functionally extreme your batches can become when the shuffling algorithm is applied to the training set.
momentum (=0.9)	Momentum in neural networks is a variant of the stochastic gradient descent. It replaces the gradient with a momentum which is an aggregate of gradients. Momentum can increase speed when the cost surface is highly non-spherical because it damps the size of the steps along with directions of high curvature thus yielding a larger effective learning rate along with the directions of low curvature.
batch_size_per_image (=1024)	The batch size defines the number of samples that will be propagated through the network. For instance, let's say you have 1050 training samples and you want to set up a batch_size equal to 100. The algorithm takes the first 100 samples (from 1st to 100th) from the training dataset and trains the network. Next, it takes the second 100 samples (from 101st to 200th) and trains the network again. We can keep doing this procedure until we have propagated all samples through of the network. Problem might happen with the last set of samples. In our example, we've used 1050 which is not divisible by 100 without remainder. The simplest solution is just to get the final 50 samples and train the network. Advantages of using a batch size < number of all samples: It requires less memory. Since you train the network using fewer samples, the overall training procedure requires less memory. That is especially important if you are not able to fit the whole dataset in your machine's memory. Typically networks train faster with mini-batches. That is because we update the weights after each propagation. In our example we've propagated 11 batches (10 of them had 100 samples and 1 had 50 samples) and after each of them we've updated our network's parameters. If we used all samples during propagation we would make only 1 update for the network's parameter. Disadvantages of using a batch size < number of all samples: The smaller the batch the less accurate the estimate of the gradient will be.
weight_decay (=0.001)	Weight decay is a regularization technique by adding a small penalty, usually the L2 norm of the weights (all the weights of the model), to the loss function. loss = loss + weight decay parameter * L2 norm of the weights. Why do we use weight decay? To prevent overfitting. To keep the weights small and avoid exploding gradient. Because the L2 norm of the weights are added to the loss, each iteration of your network will try to optimize/minimize the model weights in addition to the loss.
base_lr (=0.0003389)	In machine learning and statistics, the learning rate is a tuning parameter in an optimization algorithm that determines the step size at each iteration while moving toward a minimum of a loss function. Since it influences to what extent newly acquired information overrides old information, it metaphorically represents the speed at which a machine learning model "learns". In the adaptive control literature, the learning rate is commonly referred to as gain. In setting a learning rate, there is a trade-off between the rate of convergence and overshooting. While the descent direction is usually determined from the gradient of the loss function, the learning rate determines how big a step is taken in that direction. A too high learning rate will make the learning jump over minima but a too low learning rate will either take too long to converge or get stuck in an undesirable local minimum. In order to achieve faster convergence, prevent oscillations and getting stuck in undesirable local minima the learning rate is often varied during training either in accordance with a learning rate schedule or by using an adaptive learning rate.
max_iter (=462)	An iteration describes the number of times a batch of data passed through the algorithm. In the case of neural networks, that means the forward pass and backward pass. So, every time you pass a batch of data through the neural network, you completed an iteration. When you increase the number of iterations, you get closer to the minima and the set of optimal parameters, and these are what improve performance.

DRAFT

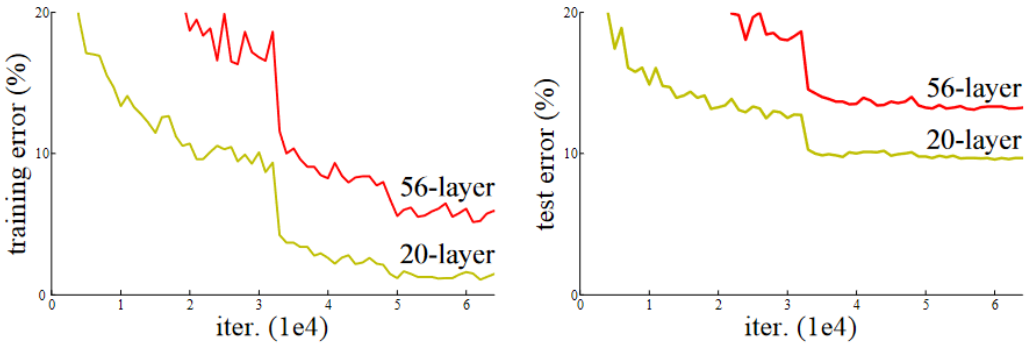

Fig. S5. These plots are taken from He et al., (2016) and they illustrate that deeper neural networks do not necessarily learn as well as shallower neural networks.

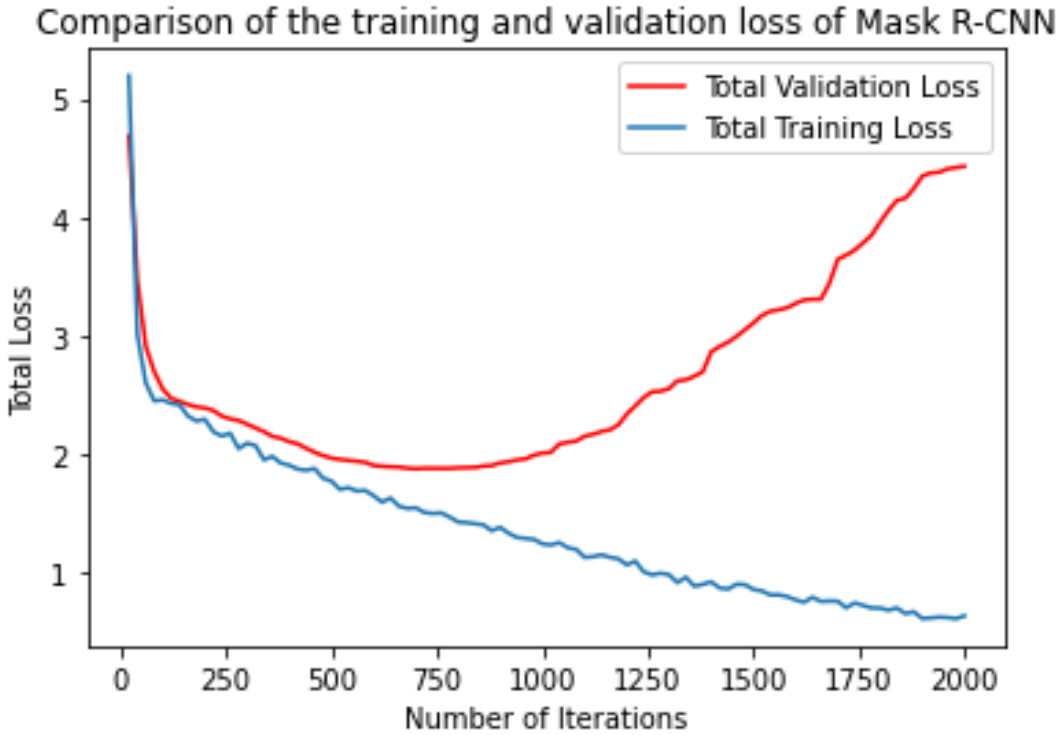

Fig. S6. The total training and validation loss of Mask R-CNN as the model trained. Both total training and validation loss were calculated every 20 iterations.

S4.1.

We found that while increasing the depth of the ResNet increased the training time of the algorithm, it improved accuracy scores. The batch size was limited by computing resources available. The number of iterations was the key metric to optimise, as we found that our model began to overfit the training data if trained for too many iterations. To determine the number of iterations, we plotted a graph (Fig. S6) of both the total training loss and the total validation loss to determine the minimum of the total validation loss, and hence the optimal number of iterations to train the model. As Mask R-CNN consists of three predictions, namely the class, bounding box and mask, the total loss is the sum of these three losses.

We see that after around 800 iterations, the model started to overfit the training data, at the expense of performance of the validation dataset. As such we determined to stop the training of Mask R-CNN at 800 iterations, and then used the saved weights after 800 iterations to predict tree crowns in Sepilok.

Supplementary Note 5: Evaluation metrics

Two key metrics are used in this work to evaluate the performance of the model: firstly, the AP50 score, and the F_1 Score. AP50 relates to the area under the precision-recall curve (AUC-PR) evaluated at a particular threshold for the Intersection over Union, in our case 0.5. A high value for AP50 means that we are seeing both high precision and recall of the model at our

particular IoU cutoff. The IoU cutoff was selected as 0.5, on the basis of previous studies evaluating the accuracy of individual tree delineation methods. Visual examples of different IoU values are given in Fig. 1.

Supplementary Note 6: Growth and mortality details

Additional results.

Details of growth and mortality.

—

Table S2. The coefficients and intercepts for the robust least squares fit between original tree height and the change in tree height.

Site	Slope	Intercept
Danum	-0.110 ± 0.006	1.90 ± 0.3
Paracou	-0.028 ± 0.0003	1.06 ± 0.01
Sepilok East	-0.060 ± 0.004	1.64 ± 0.11
Sepilok West	-0.072 ± 0.003	1.68 ± 0.12

DRAFT