University of Nevada, Reno

Improving Tree Crown Mapping using Airborne LiDAR with Genetic Algorithms

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Science in Natural Resources & Environmental Science

by

Johanson Onyegbula

Dr. Jonathan Greenberg/Thesis Advisor

August 2023

THE GRADUATE SCHOOL

We recommend that the thesis
prepared under our supervision by

entitled

be accepted in partial fulfillment of the
requirements for the degree of

Advisor

Committee Member

Graduate School Representative

Markus Kemmelmeier, Ph.D., Dean
Graduate School

Abstract

Landscape-scale mapping of individual trees derived from LiDAR (Light Detection And Ranging) data have been found to be valuable for a wide range of environmental analyses including carbon inventories; fuel estimations for wildfire risk assessment and management. These mapping efforts use individual tree crown (ITC) recognition algorithms applied to LiDAR point clouds, which have complex parameter sets. Genetic algorithms (GA) have been demonstrated to be excellent function optimizers for very complex search spaces and perform well for parameter tuning. Here, we use GAs to identify the best of a set of published ITC models and their optimal parameters for airborne LiDAR of forested plots in the Sierra Nevada Mountains of California. We assessed the accuracy of these ITC models in terms of the F-score and percentage bias for tree crown prediction. GA-optimization generally improved on ITC default parameters and showed that these models typically perform better for detecting overstory trees.

Introduction

Mapping tree crowns at an individual level over large scales can be useful for estimating leaf area (Ke et al. 2016) and canopy cover (Carreiras et al., 2006), which can be used for making inferences about light penetration and availability, microclimates and understory competition (Binkley, 2021). Also, accurate individual tree maps at landscape levels are useful inputs into wildfire and insect outbreak models (Sánchez et al., 2018; Negron et al., 2009). The more precise these maps are, the better models we can develop for mitigating these issues.

Mapping individual trees at a landscape level is often slow and infeasible for conventional field techniques (Ko et al., 2009). Hence, we need a way of acquiring these data over large and often inaccessible regions reliably and quickly. Remote sensing, specifically airborne LiDAR (Light Detection and Ranging), produces datasets conducive for doing these landscape-level analyses.

Over the previous decades, multiple individual tree crown (ITC) recognition algorithms have been developed that can be applied to LiDAR datasets for the purpose of producing landscape-level forest inventories at the scale of individual trees (Wulder et al., 2000; Dalponte et al., 2016; Silva et al., 2016; Li et al., 2012; Vega et al., 2014; Hamraz et al., 2016). These algorithms often have complex parameter sets that may not necessarily have real-life analogs (i.e., cannot be measured in the field) that, if poorly tuned, can significantly degrade the performance of the ITCs for tree crown mapping. Hence, we are posed with the challenge of optimizing their parameters for the best possible performance.

Multiple techniques exist for optimizing the values of parameters in a search space. Hill climbers are a common technique for exploring a single solution at a time but are prone to getting stuck in local optima and are not suited for discrete, poorly understood search spaces (Johnson and Jacobson, 2002). Moreover, multiple combinations of parameters for these ITCs can yield similar levels of accuracy for tree crown mapping, thereby increasing the chances of hill climbers to find only local optima.

Machine learners have seen recent advances in various spheres but often require training datasets with inputs and known possibilities as labels. However, these are often unsuitable for tree crown mapping as only inputs without corresponding output labels (tree crown shapes) are typically available. Neural networks in particular, which are suited for this likely computer vision problem require huge datasets which are unavailable in this context (Shaikhina and Khovanova, 2017).

Genetic algorithms (GAs) overcome the challenges of the previous optimization techniques as they work well with sparse datasets and are suited for discontinuous, multi-modal, poorly understood search spaces. GAs are evolutionary algorithms that utilize the principle of natural selection in a mathematical context for evolving solutions over multiple iterations or generations (Goldberg, 1988). They encode parameters directly in bit strings as chromosomes and are effective for this kind of problem as they seek to maximize a defined fitness value representing the context of what is to be optimized.

Here, GAs are used for simultaneously optimizing the parameters from a set of widely available ITC models with the goal of selecting the best for landscape tree crown mapping of airborne LiDAR. The tree crowns are validated using tree stems acquired for

corresponding LiDAR plots and the performance of the GA-tuning is evaluated using various tree level and plot level metrics. The performance of the GA-optimized parameters is then compared to their default values. Finally, we assessed their performance over various DBH cutoffs.

References

Binkley, D. (2021). Forest ecology: an evidence-based approach. John Wiley & Sons.

Carreiras, J. M., Pereira, J. M., & Pereira, J. S. (2006). Estimation of tree canopy cover in evergreen oak woodlands using remote sensing. Forest ecology and management, 223(1-3), 45-53.

Dalponte, M., & Coomes, D. A. (2016). Tree-centric mapping of forest carbon density from airborne laser scanning and hyperspectral data. Methods in ecology and evolution, 7(10), 1236-1245.

Goldberg, D.E. (1988). Genetic Algorithms in Search Optimization and Machine Learning.

Hamraz, H., Contreras, M. A., & Zhang, J. (2016). A robust approach for tree segmentation in deciduous forests using small-footprint airborne LiDAR data. International Journal of Applied Earth Observation and Geoinformation, 52, 532–541. https://doi.org/10.1016/j.cageo.2017.02.017

Johnson, A. W., & Jacobson, S. H. (2002). On the convergence of generalized hill climbing algorithms. Discrete applied mathematics, 119(1-2), 37-57.

Ke, L. I. U., ZHOU, Q. B., WU, W. B., Tian, X. I. A., & TANG, H. J. (2016). Estimating the crop leaf area index using hyperspectral remote sensing. Journal of integrative agriculture, 15(2), 475-491.

Ko, D., Bristow, N., Greenwood, D., & Weisberg, P. (2009). Canopy cover estimation in semiarid woodlands: comparison of field-based and remote sensing methods. Forest Science, 55(2), 132-141.

Li, W., Guo, Q., Jakubowski, M. K., & Kelly, M. (2012). A new method for segmenting individual trees from the lidar point cloud. Photogrammetric Engineering & Remote Sensing, 78(1), 75-84.

Negron, J. F., McMillin, J. D., Anhold, J. A., & Coulson, D. (2009). Bark beetle-caused mortality in a drought-affected ponderosa pine landscape in Arizona, USA. Forest Ecology and Management, 257(4), 1353-1362.

Sánchez Sánchez, Y., Martínez-Graña, A., Santos Francés, F., & Mateos Picado, M. (2018). Mapping wildfire ignition probability using sentinel 2 and LiDAR (Jerte Valley, Cáceres, Spain). Sensors, 18(3), 826.

Shaikhina, T., & Khovanova, N. A. (2017). Handling limited datasets with neural networks in medical applications: A small-data approach. Artificial intelligence in medicine, 75, 51-63.

Silva, C. A., Hudak, A. T., Vierling, L. A., Loudermilk, E. L., O'Brien, J. J., Hiers, J. K., Khosravipour, A. (2016). Imputation of Individual Longleaf Pine (Pinus palustris Mill.) Tree Attributes from Field and LiDAR Data. Canadian Journal of Remote Sensing, 42(5), 554–573. https://doi.org/10.1080/07038992.2016.1196582.

Vega, C., Hamrouni, a., El Mokhtari, S., Morel, J., Bock, J., Renaud, J.-P., … Durrieu, S. (2014). PTrees: A point-based approach to forest tree extraction from lidar data. International Journal of Applied Earth Observation and Geoinformation, 33, 98–108. https://doi.org/10.1016/j.jag.2014.05.001

Wulder, M., Niemann, K. O., & Goodenough, D. G. (2000). Local maximum filtering for the extraction of tree locations and basal area from high spatial resolution imagery. Remote Sensing of environment, 73(1), 103-114.

Acknowledgements

My greatest appreciation is to God Almighty, for His immense graciousness in my life, and the strength and good health granted to me to execute this project.

Immense gratitude also goes to my advisor, Dr. Jonathan Greenberg, who chose to work with me for my masters, for his guidance, style of supervision which ensured my work-life balance, clearly defined expectations, and support in ensuring the success of the execution of this project. And also my master's committee members: Dr. Erin Hanan and Dr. Alireza Tavakkoli.

I would like to acknowledge Theodore Hartsook and Adriano Matos whose help I needed more often than any other student for my thesis. My gratitude also extends to Professor Sushil Louis, the genetic algorithms professor whose class I took in my very first semester at the University of Nevada, Reno; the concepts from the class and his ideas afterwards largely shaped my thesis.

Finally, I cannot neglect the importance of my parents and three sisters, whose constant support, love and prayers played a key role in my life and education from childhood till date. May your efforts never be in vain. Amen!

Table of Contents

List of Tables

List of Figures

List of Equations

Preface

This chapter is in preparation for submission to two journals: Remote Sensing of Environment and IEEE Transactions on Geoscience and Remote Sensing.

Chapter 1 – Improving Tree Crown Mapping using Airborne LiDAR with Genetic Algorithms

Abstract

Landscape-scale mapping of individual trees derived from LiDAR (Light Detection And Ranging) data have been found to be valuable for a wide range of environmental analyses including carbon inventories, inputs into tree growth models, and fuels estimations for wildfire risk assessment and management. These mapping efforts use individual tree crown (ITC) recognition algorithms and several ITCs have been developed for delineating trees from point cloud. These ITCs have complex sets of parameters and vary considerably in their ability to detect trees under varying forest stand sizes, canopy layers and density conditions.

Genetic algorithms (GA) have been demonstrated to be excellent function optimizers for very complex search spaces and have been shown to perform well for parameter tuning. Here, we use GAs to identify the best ITC model, out of 6, and its optimal parameters for airborne LiDAR of forested plots. We used a fitness function which focused on balancing the number of predicted tree crowns and their positions relative to actual tree stems. We assessed the accuracy of these ITC models and GA-optimized parameters in terms of the F-score, percentage bias and other metrics; then compared the performance GA-optimized parameters to the ITC defaults their default parameters to assess the degree to which the genetic algorithm improved on the performance of naive implementations of the ITC algorithms. GA-optimization generally improved on ITC default parameters, and we also showed that these models generally perform better for detecting overstory trees.

Introduction

Individual tree mapping has gained increased focus as modern techniques have moved from manual to various forms of automated mapping methods (Karlson et el., 2014; Duncanson et al., 2014). Often, landscape-scale mapping of trees at a generic level is done and the data is typically used for assessing percentage of tree cover on land, classifying vegetation type, monitoring forest conversion, leaf area and canopy closure (Heumann, 2011). However, advances in technology, individual and government needs have necessitated the increased relevance of individual tree mapping at these landscape-scale levels that were previously generalized.

Deriving information on trees at an individual level is often done to identify and aggregate various tree species, which can be useful for assessing the vulnerability of the associated flora and fauna (Bunting et al., 2010) especially as increased deforestation happens in many industrialized parts of the world. Assessing tree stem volume for estimating carbon stock based on CO_2 sources and sinks is becoming increasingly useful today, as many countries today seek to reduce their carbon footprints (Tonolli et al., 2011). Such information comes from individual tree properties including tree height, diameter and basal volume, which require data at fine resolutions to be estimated. And unlike multiple studies on tree stem mapping which are typically done in small areas (Zhang, 2019; Qian et al., 2019; Liang et al., 2011), acquiring such information at a landscape level is useful for assessing the potential impact of forested environments on the climate.

Beyond just individual tree attributes, the underlying spatial distribution of these forests is essential for wildfire mitigation and insect outbreak analysis (Forbes et al., 2022; Lantschner and Corley, 2015). Understanding the demography of trees in a forest is also useful for monitoring and making projections of forest productivity. And such analysis often relies on attributes about size classes, regeneration gaps, productivity and mortality (Ogden, 1985). These useful details for assessing forest dynamics are only accessible with tree level characteristics at a landscape scale. Monitoring forest growth, age, growth and the relationship between various species including competition and dominance are important steps for assessing forest demography and making useful predictions (Enright and Hartshorn, 1981).

Tree species identification is also useful in assessing forest ecosystem functionality as forest functionality typically benefits more increased biodiversity. These tree crown maps can also help determine the dominant tree species in a forest (Queinnec et al., 2023), which are significant determiners of floristic turnover (Draper et al., 2019). Furthermore, monitoring tree growth (and eventual forest growth); calculating and monitoring tree biomass; and carbon flux is very important for assessing the health of forests and implications on the climate today (Zhao et al., 2018). Tree crown mapping, which is useful for determining tree canopy volume, can be used to inform decision making for various management purposes in agriculture (Marín-Buzón et al., 2020).

Determining all these tree-level attributes at a landscape scale is important for fast, accurate and representative information on forest attributes which are also useful for agricultural health monitoring (Ji-Hua and Bing-Fang, 2008), forest fire mitigation and

treatment plans (Li et al., 2022), insect outbreak monitoring (Fassnacht et al., 2014) among others. The vertical and horizontal continuity and gaps between understory and overstory vegetation affect the spread of any potentially destructive agent such as wildfires, as increased fuel loading typically increases the wildfire severity (Hanan et al., 2021). Consequently, precise spatial maps, especially in 3D forms can serve as a basis for developing more reliable scientific models for analyzing and mitigating these naturally occurring disasters in forests.

Historically, individual tree mapping utilized laborious, manual field surveys in which trees had to be geolocated usually through the establishment of reference positions and determine the range and azimuth from these positions. The amount of effort to map individual trees using field surveys scales fairly directly with the number of trees or areas under investigation, so landscape-scale surveys are impractical using these techniques. Moreover, it is nearly infeasible to precisely map tree crowns with fieldwork. The use of remote sensing data, particularly data collected at resolutions significant finer than the size of individual tree crowns (Brolly and Kiraly, 2009; Lindberg and Holmgren, 2017), have been proposed as an alternative to field surveys for creating landscape-scale stem maps. These techniques have employed the use of unmanned aerial vehicles (UAVs) (Schiefer et al., 2020; Onishi and Ise, 2021); terrestrial (Tao et al., 2021) and airborne (Li et al.; 2012) LiDAR (Light Detection and Ranging); multispectral (Abutaleb et al., 2021) and hyperspectral satellite imagery (Shafri and Hamdan, 2009). However, these techniques introduce new challenges of producing stem maps with sufficient accuracy to replace field surveys.

A common challenge with tree crown mapping involves the spatial resolution of such maps, which is a significant consideration for the domain of application of the final product (Nwilo et al., 2022; Greenberg et al., 2009). The visibility from lighting conditions and weather constraints imposes severe limitations on data from passive remote sensing sources such as satellite imagery and UAVs (Khanal etal., 2020). Ethical issues concerning data access and ownership in remote sensing can often arise (Slonecker et al., 1998; Sanger and Barnett, 2021), and this is important with regards to adjoining private agricultural sites considering large swaths are typically captured in remotely sensed data. Occlusion due to cloud cover (Ajayi and Ojima, 2022); aerosols and smoke (Ulku et al., 2022); and overstory trees for subcanopies (Ryu et al., 2014) are challenges that need to be addressed for effective tree crown mapping with remote sensing imagery. Also, fine tree level attributes beyond the canopy like height and diameter at breast height (DBH) are often difficult to estimate, and these are relevant to reconstruct the forest scene and delineate trees (Morsdorf et al., 2004).

Amongst the various datasets used in individual tree crown recognition, small footprint airborne laser scanning (ALS) otherwise known as airborne LiDAR, has a great track record of producing accurate stem maps (Tang et al., 2015; Popescu, 2007; Næsset and Bjerknes, 2001). LiDAR is an active remote sensing technology that utilizes laser pulses for capturing information about the environment. Point clouds produced from LiDAR are effective for computing the absolute/relative distances and heights of objects (Kellner et al., 2019), a quality unobtainable with multispectral, high spatial resolution datasets. Also, the LiDAR emitted pulses typically have multiple returns and can hence provide

profile information at various elevations (Dong and Chen, 2017), yielding accurate data about sub-canopy layers in a forest (Renslow et al., 2000).

However, passive remote sensing is severely limited in extracting such detailed spatial information, even at very fine spatial and radiometric resolution due to the single pixel value in output rasters. Technologies such as LiDAR provide the ability to obtain such levels of detailed spatial information at sub-canopy levels, due to records from multiple returns of a single laser pulse to the same location. Hence, even more precise spatial maps of trees, rather than an aggregation of forest cover, is available with LiDAR. Terrestrial LiDAR from terrestrial laser scanners (TLS) in particular, can provide detailed information about tree stems and the vertical profile of individual trees, synonymous to the perspective of a person standing in front of a tree. The visibility of details captured by TLS for tree crowns can be obscured in dense forests or where tree continuity in the horizontal and vertical dimensions is locally emphasized (Zong et al., 2021). However, airborne LiDAR tends to capture better tree crowns as its sampling footprint is typically larger than TLS and multiple returns tend to capture various canopy layers until the very last return – ground.

The use of satellite imagery for tree crown mapping is often constrained by the spatial and radiometric resolution of the imagery, which often requires feature identification and classification of fine details to detect individual trees. Google Earth imagery has been employed in tree crown mapping which was fed into a region-based convolutional neural network (CNN) that achieved over 80% accuracy for both number of trees and crown area (Yang et al., 2022). Very fine spatial resolutions aid individual tree crown

recognition with this type of imagery as seen in Karlson et el. (2014) where Geographic Object Based Image Analysis (GEOBIA) was used with Worldview-2 image for delineating trees, with accuracy level positively correlated with tree crown area. Multispectral satellite imagery is sparsely used in isolation for tree stem mapping as available resolutions are often not feasible for individual tree detection (Larsen, 2007) and often require integration with a more precise and detailed remotely sensed data source such as LiDAR (Holmgren et al., 2008).

Hyperspectral imagery typically provides a more reliable means of mapping individual trees than multispectral imagery, as seen in Moriya (2021), where it was used in mapping infected citrus trees with an accuracy of 94% and F-score of 0.85 compared to just multispectral imagery with similar respective results of 79% accuracy and 0.55 as the F-score. Bark beetle damage was also mapped at the tree level with hyperspectral imagery in Näsi (2015) with 74.7% accuracy at the single tree level for trees with DBH above 26.2cm and at least 23.9m tall. Unfortunately, the accuracy of remotely sensed imagery is severely limited due to the inability to detect understory trees. Hence, they are best suited for isolated trees in single vertical layers when solely used or combined with LiDAR for detailed information (Alonzo, 2016).

Due to the level of detail provided by the data and its ability to map understory trees in dense forests, LiDAR remote sensing is being increasingly used for individual tree mapping. In Brolly and Kiraly (2009), TLS was used to map individual trees at a forest reserve, which was used to derive individual tree heights, DBH and their locations. Another study by Liu et al. (2017) employed the use of matching for multiple TLS scans

of a single forest plot to reconstruct and subsequently map tree stems. TLS stem mapping algorithms generally use point clustering and geometric analysis to identify tree stems. One common technique of mapping individual trees from TLS involves the generation and extraction of ground points, and the classification of remaining points for modelling stems based on the connectedness of points (Brolly and Kiraly, 2009). Another algorithm employed for tree detection from TLS point cloud involves stem identification and isolation based on connected component segmentation and curvature of points (Zhang, 2019).

A common approach to tree segmentation algorithms from airborne LiDAR employs the use of local maximum filtering and slope breaks for detecting individual trees from a rasterized canopy height model (CHM) as seen in the watershed algorithm developed in Wulder et al. (2000). This study used either a fixed or dynamic window size with omnidirectional transects from a central pixel until a minimum radiance value is reached. Bigger trees were easier to detect than smaller ones, the latter which were more likely to be detected with smaller window sizes. Dalponte and Silva also use this similar local maximum technique to detect trees from a CHM, designating these maxima as tree tops to isolate individual trees, and having height thresholds for what considering pixels would be used for classifying a cluster of points as a single tree.

Holmgren (2008) also utilized this raster-based technique for Individual Tree Crown (ITC) delineation, and the watershed algorithm is a quite popular method of mapping trees available in R (R Core Team, 2022). Ayrey et al. (2017) uses layer stacking in a slight variation from the CHM model for isolating trees. This layer stacking algorithm

attempts to overcome the previous challenge of raster-based ITCs by slicing the entire forest into 1m intervals and attempting to cluster points in each of these sliced layers. Afterwards, it attempts to stack these layers to find cluster location agreements as representations of individual trees.

There are also non-raster-based methods of segmenting trees – each with its unique strengths and weaknesses. These segmentation algorithms all have their unique parameters with varying implications that directly affect the performance on LiDAR datasets. An integration of the watershed's CHM model and raw LiDAR returns was employed in Duncanson et al. (2014) for multi-layer tree-level classification, which was used for detecting understory and overstory trees. And although it had difficulty detecting small, dense understory trees, it performed well in identifying trees of various strata – dominant, co-dominant and intermediate-level trees at decreasing levels of accuracy.

Li et al. (2012) describes a tree delineation method that is based on raw point cloud, which identifies trees based on the relative spacing of points between trees after height normalization based on a defined threshold distance. It works well where the spacing between trees at the top is greater than the spacing at the bottom, typical in coniferous forests. Ptrees (Vega et al., 2014) is another tree segmentation algorithm that works with raw point clouds and uses the raw elevation values (rather than normalized height) to perform multiple segmentations at different scales before selecting the best set of apices that are used for the final segmentation. It achieves this while preserving the geometry of crown points and similarly works better for conifers than deciduous plants, and tends to have higher commission errors than other methods that work with raw point cloud.

In Ferraz (2012), an iterative algorithm was developed that works directly with the point cloud segments trees based on a selected kernel bandwidth and uses the mean-shift algorithm. This algorithm is based on the assumption that the point cloud from airborne LiDAR consists of multiple modes, and hence the goal is to extract these modes which represent a tree. It works well for multi-layer forests and adequately differentiates between ground vegetation and taller trees with up to 98.6% detection for dominant trees. Hamraz et al. (2016) is a non-parametric approach that uses the highest elevation points in a gridded LiDAR point cloud, which is then expanded out and used to identify trees based on identified crown boundaries. Afterwards, elevation points in each grid below a threshold height are discarded and a convex hull of boundary points are also used to generate tree crowns. It detected 100% of all dominant trees and decreased in detection ability for trees of lower strata with an overall accuracy of 77%, while omissions were generally more common than commission errors. A common hassle with utilizing and comparing the relative performances of these segmentation algorithms involves the determination of the optimal value of their parameters.

Delineating individual trees from remote sensing data often depends on several parameters, and the optimal selection of these parameter values is often a common challenge remote sensing analysts have to overcome (Ke and Quackenbush, 2011). A common approach for delineating trees with passive remote sensing imagery relies on finding the local maxima of pixels, based on the assumption of tree crown centroids coinciding at that point (Culvenor, 2002). The detection of local optimas of groups of pixels is a technique exploited by classification algorithms in machine learning for

delineating individual trees from imagery, alongside various geometric properties of trees (Yu et al., 2006).

Feature selection often poses a major computational challenge in accurately delineating trees and making selections among the vast array of features to be optimized can be quite challenging (Kattenborn, 2021) for even experts. And although machine learning approaches might have experienced success in many tree detection applications, it often requires a lot of transformation of data to extract reasonable parameters and supervised learning, which requires human guidance, is typically the preferred approach (Weiss et al., 2010). This would pose a major challenge of addition parameter and hyperparameter inclusion when tuning several models, each of which have their own parameters. Moreover, parallelization of machine learning models is a complex issue (Cheng, 2021) and these classification algorithms work best on continuous, mathematical spaces with differentiable parameters. Hence, a different approach is needed to optimize discrete parameters of multiple models in parallel.

Genetic algorithms (GAs) are an evolutionary-based approach to effectively exploring search spaces and are based on the principle of natural selection or a “survival of the fittest” approach in their exploration (Goldberg, 1988). It involves the use of a parallel population of individuals, also known as chromosomes, encoded with the parameters whose optimal values are being sought. They are highly effective in complex search spaces and have found great applications in optimization problems (Louis and Johnson, 1997; Damanik et al., 2019; Liashchynskyi and Liashchynskyi, 2019). The interpretation of optimality in this exploration largely rests on the specific problem. Each individual

explores the complex search space with various fitness values assigned to them, based on a mathematical metric designed to assess their performance, to converge at an optimal solution.

A pre-assigned probability of crossover determines if two individuals are selected to exchange information, and such combination is often directly or indirectly proportional to the fitness of individuals. Over various generations, the fitness of these individuals tends to improve and converge to the values of the fittest chromosomes with typically optimal values of parameters achieved. A typically small probability of mutation helps to preserve the diversity of individuals in their exploration and hence prevents premature convergence at a local optimum. GAs are highly effective in quickly exploring complex search spaces, and take significantly less time than an exhaustive search to converge at an optimal solution by exploring patterns of fitness (Goldberg, 1998).

Fitness values are often mathematical functions whose solutions yield the maximum value of the function, hence are always maximized. Simple GA problems often search through the space of solutions for the solution to one or multiple variables. Multi-objective fitness functions are quite common in GA-problems where multiple goals are sought like where health and damage done to an opponent are being simultaneously optimized, or in the design of aircrafts where drag is being minimized and lift is maximized. The aim of such multi-objective functions is to optimize objectives in various dimensions (Paterna et al., 2020). This can be as common as pareto-optimality problems where a solution is only better than a previous one if it better optimizes all its "objectives" than another (Horn et al., 1994). A pareto-optimal objective function was

used when estimating plasma and temperature gradients in X-ray spectroscopy (Golovkin et al., 2000), while a cumulative objective function was used in decision-making for a multi-use reservoir serving both human and ecological needs (Chang et al., 2010).

Many studies have also incorporated genetic algorithms into optimization problems and found the solutions much better than those of conventional optimization techniques that have been previously used and continuously developed (Lin and Zhang, 2020; Marchesini and Farinelli, 2020). Gesù and Bosco (2005) described an image segmentation technique based on GAs in which the image segmentation problem was defined and reduced to that of an optimization problem, and outperformed 3 other image segmentation algorithms previously used, and had a faster convergence to the correct solution. Mumtaz et al. (2008) assessed the relative performances of an existing image fusion algorithm and three image fusion algorithms in which GAs are used. And despite the computational complexity of the GA-based image fusion algorithms and subsequent unsuitability for real-time applications, they were shown to outperform the existing non-GA integrated technique and proved suitable for static image fusion applications.

Aim and Objectives of Research

Tree crown mapping is useful for monitoring and informing decision making about tree cover on land, vegetation type, monitoring forest conversion, leaf area and canopy closure (Heumann, 2011). With airborne LiDAR, we can delineate these individual tree crowns precisely and various ITCs with complex sets of parameters have been developed for which we seek to improve on, by optimizing the discrete parameters each provides, for mapping trees.

Hence, if a complex search problem can be reduced to an optimization goal, it can be solved with a genetic algorithm (Goldberg, 1988). Consequently, the aim of this study is to improve tree crown mapping of forested regions with airborne LiDAR. The following research questions are the objectives we seek to answer:

1. Can we optimize the performance of ITC recognition algorithms for tree crown mapping?
2. How do our GA-optimized parameters compare to the default values?
3. How does the performance of ITC recognition algorithms vary for different tree DBH size classes?

Methods

For this study, airborne LiDAR data, for which tree crown recognition would be implemented on, was acquired over various plots. Also, TLS data for the same plots were acquired to be used as ground truth for validating results. Thereafter, various ITC recognition algorithms across a range of parameters about their default values were tested on a few plots and the results were used in designing the fitness functions for the genetic algorithms that would possibly optimize tree crown recognition. These GAs were run on training plots and the results were validated on test plots that were withheld prior to any computation. Afterwards, these were evaluated with various accuracy metrics on various DBH cutoffs and used to map trees over multiple plots. This is shown in Figure 1.

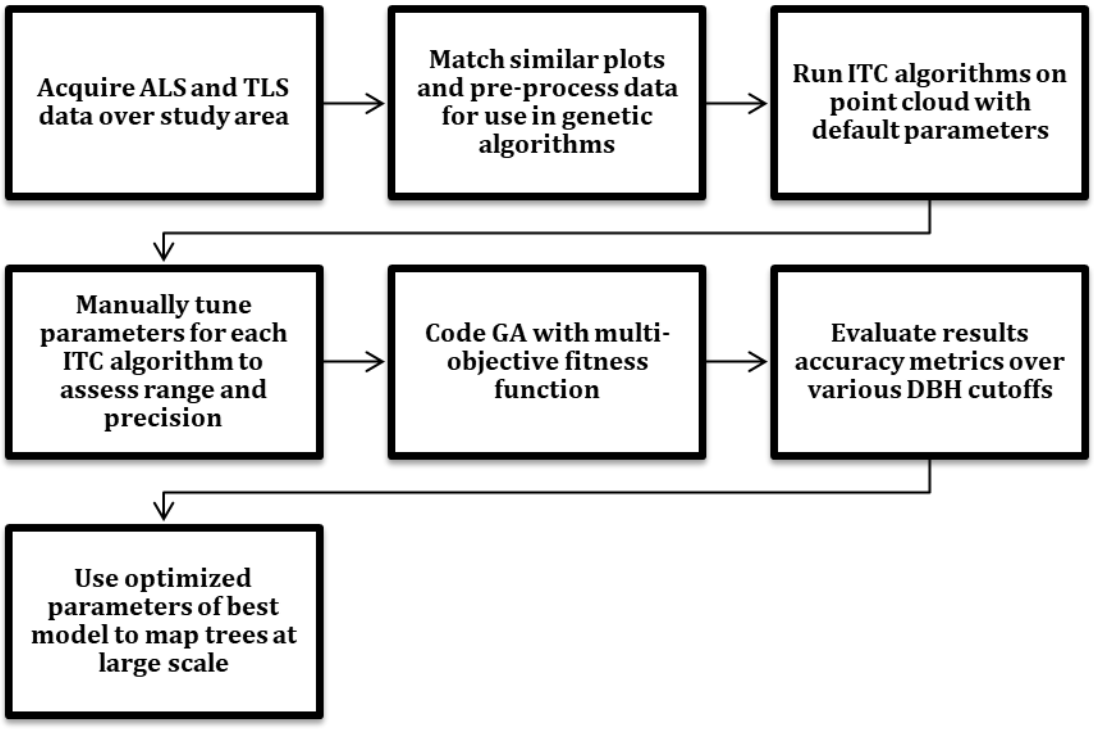


Figure 1: Flowchart representing workflow of the entire study.

Study Area and Plot Design

The study area is a forested region of the Sierra Nevada Mountains, CA/NV, specifically the Lake Tahoe Basin Management Unit (LTBMU) and Plumas National Forest (PNF). The LTBMU is responsible for the management and preservation of the watershed ecosystem of Lake Tahoe, which is a popular destination in a high-risk fire environment. While the Plumas National Forest spans has mostly old-growth forests.

Both sites in Figure 2 are mixed conifer forests in a Mediterranean climate with elevation ranging from 215m to 2,818m above sea level. They are characterized by long, hot-dry summers and cool, wet winters with increasing precipitation at higher elevations (Moody et al., 2006). Current management efforts aim at protecting these sites from wildfires, which have been altered by fire suppression over the last 100 years, after an early period of clear-cutting up until the 20th century (Stanton and Dailey, 2007). Figure 2 is a map of the study area produced with ArcMap (Esri Inc., 2020).

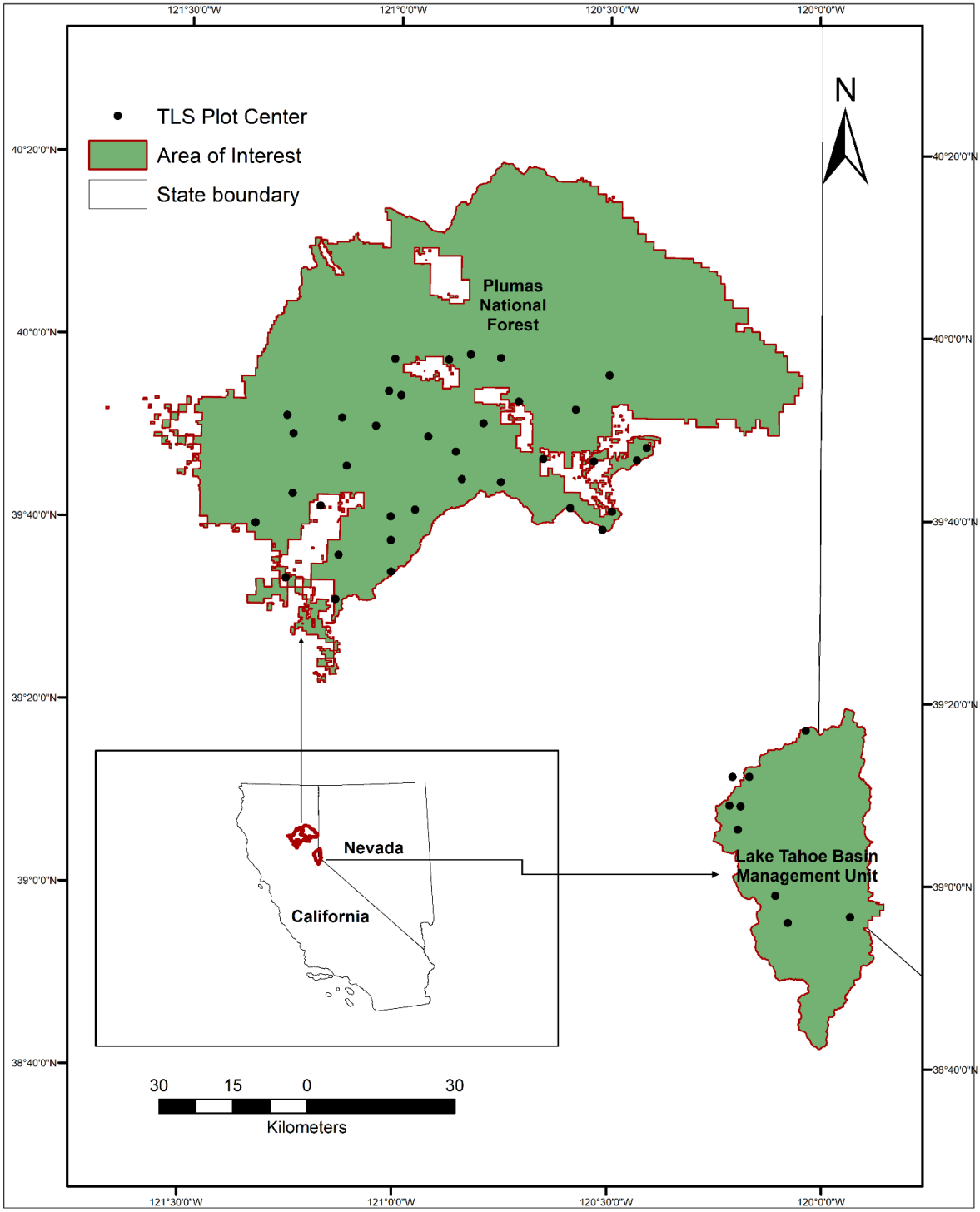


Figure 2: Study area with TLS Plot centers shown

Data Collection and Pre-processing

Airborne LiDAR: Airborne LiDAR data was acquired over the entire Plumas National Forest and the Lake Tahoe Basin Management Unit (Table 1). The area was tiled into square plots in the study area of length 30m, with a 5m buffer about the edges. Hence, each airborne LiDAR plot had point clouds for areas spanning approximately 40m by 40m in size, which were pre-processed for use as input data.

In addition to the airborne LiDAR data acquired over the study area, TLS data was acquired for selected plots in the same region. The TLS data served as ground truth data for validation.

Calibration/Validation Data: Our analysis required positions and sizes of stands of trees confined to specific areas (plots). To accomplish this, we leveraged a plot network with a two-stage stratified random sampling design to ensure a representative sample of forest types across Plumas National Forest and the Lake Tahoe Basin. The first stage of sampling involved placing 10 km grid cells within the boundaries of the sample sites. The second stage placed a 30 x 30 m plot centered within each grid cell randomly selected to be on publicly accessible land, between 60-120 m from the nearest road, and have more than 10% tree cover based on a LANDSAT analysis and were stratified by location. Also, plot locations in any areas with recent burn scars or signs of recent logging were avoided (Wade, 2022). A total of 108 plots were scanned and used in this analysis during leaf-on periods in 2017 and 2018. Each plot represents a 30 x 30 m square (plus a 5 m buffer) centered on the plot center using 9-13 separate scanner positions, with 2-4 scans per position (dependent on stand density) to maximize the vertical distribution of data (Figure

3). Additional scan positions were added to maintain a visual line of sight between reflectors. Scans were taken with a Riegl VZ-400i Terrestrial Laser Scanner (Table 2). All scans were coregistered within-plot using RiSCAN PRO (Riegl Laser Measurement Systems GmbH, 2020), and then subsequently to an airborne laser scanning (ALS) base map using LAStools (RapidLasso GmbH, 2019), and CloudCompare (Girardeau-Montaut, 2020). The average number of points per plot was 125,830,459 and the average density was 7.86 points/cm^2 (Hartsook, 2021).

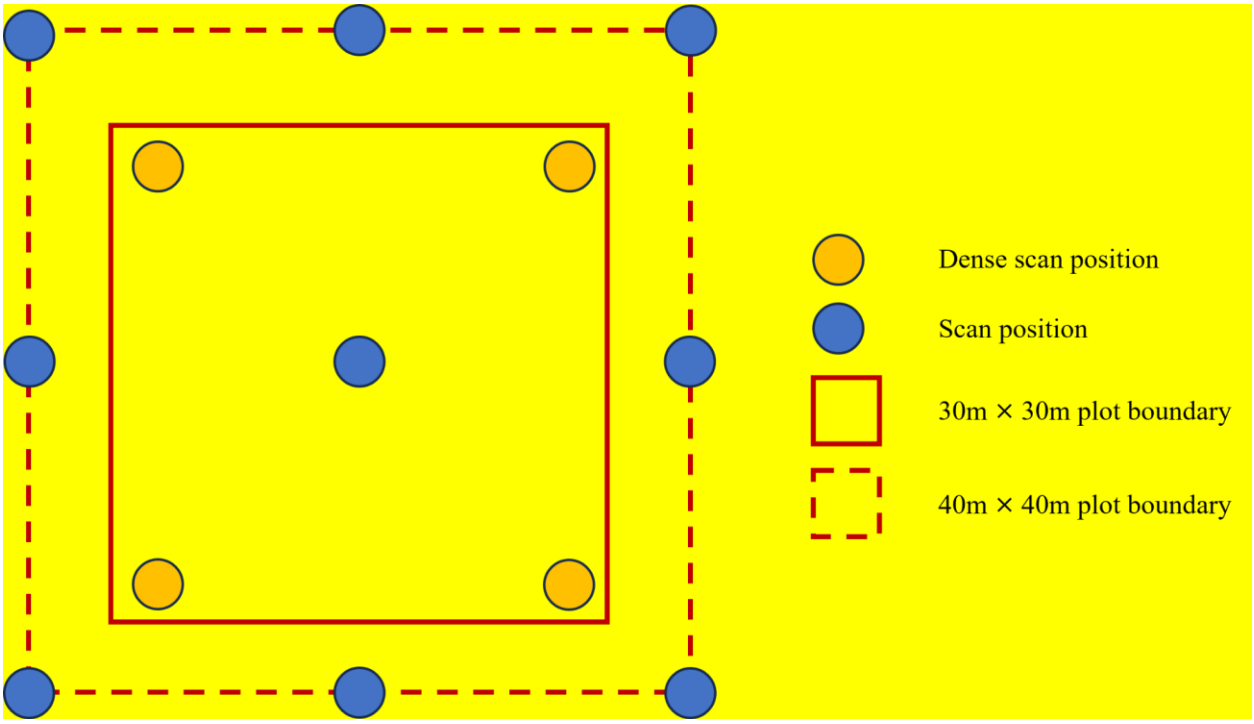


Figure 3: TLS scan set-ups (5m buffers were used to prevent edge effects)

The airborne LiDAR data was acquired with the specifications shown in Table 1, while Table 2 shows the specifications for which the TLS data was acquired.

Table 1: Airborne LiDAR flight specifications

Multi-Swath Pulse Density	≥ 8 pulses/m^2
Field of View (Scan Angle)	$< 30^o$ ($\pm 15^o$ from Nadir)
Returns/Laser Pulse	Up to 4
Laser Footprint Diameter	20-25 cm
Intensity Range	1-255
Swath Overlap	$\geq$ 50% side-lap (100% overlap); up to 75% in some areas
Pulse rate	$\geq$ 150,000 laser pulses per second
Data Completeness	No voids
Maximum GPS Baseline	$\leq$13 nautical miles
GPS PDOP During Acquisition	≤ 3.5
GPS Satellite Constellation	≥ 6
Vertical Accuracy (1.96σ), bare earth, slope $< 20^o$	< 30 cm
RMSE (1σ)	$\leq$ 15 cm
Horizontal Accuracy (1σ), bare earth, slope $< 20^o$	< 30 cm
Between-swath reproducibility	$\leq$ 15 cm
Range reproducibility	$\leq$ 5 cm

Table 2: Terrestrial LiDAR acquisition specifications

Laser scanner	RIEGL VZ-400i
Pulse repetition rate	1.2 MHz
Data acquisition speed	500,000 measurements per second
Range	800m
Accuracy	5mm
Precision	3mm

Laser wavelength	Near infrared
Laser Beam Divergence	0.35 mrad
Laser Class	Class 1
Field of view	100^o (Vertical $+60^o$ / -40^o) × 360^o (Horizontal)
Vertical scan speed	3 – 240 lines per second
Horizontal scan speed	0^o to 150^o per second
Angle measurement resolution	2.5 arcsec (Vertical) and 1.8 arcsec (Horizontal)
Cloud connectivity	Wi-Fi and 3G/4G LTE
Compatibility	RIEGL VMZ Mobile Laser Mapping System
Internal storage	256 GB SSD (Solid State Disc)

For the TLS to be used for validation against the airborne LiDAR point cloud, a means of simplifying the overlaying and comparison of delineated trees against tree stems was needed. Hence, rasters were generated from 5cm vertical slices about 1.37m height (approximate DBH) of the TLS data. Afterwards rasters of 1cm spatial resolution were generated from the point cloud between 1.345 – 1.395m (5cm depth) which generally carved out near-circular stem outlines in the raster. These was done in an automated workflow with pdal (PDAL Contributors, 2022). These rasters were then digitized and converted to shapefiles for use for validation as tree stems in QGIS (QGIS.org, 2019). Attributes of the shapefiles were mostly about tree stems with attributes including diameter at breast height (DBH), species, azimuth and (rarely) short notes about the trees.

Corresponding airborne LiDAR and TLS plots were matched and cropped within the extent of the generated raster using the raster package (Hijmans, 2022). Forty-five

matching plots were identified, for which there was an 80/20 training and test set split. The 45 airborne LiDAR plots which had an average density of 28.18 points/m², were split into 3 distinct point cloud density categories of high, medium and low densities based on the total number of returns per plot – 15 per category. For each category, 12 were randomly selected as training plots and the remaining 3 were withheld as test plots. Thirty-six corresponding plots were used for training and hence used in the actual genetic algorithms for parameter tuning, and 9 for validation after GA-fine tuning. Table 3 below shows the distribution of DBH of trees in all plots.

Table 3: Statistics on DBH distribution of trees in the training and test plots used

	Min DBH (cm)	**Max DBH (cm)**	**Under 5cm (%)**	**5 – 10 cm (%)**	**10 – 15 cm (%)**	**15 – 20 cm (%)**	**20 – 25 cm (%)**	**Over 25cm (%)**	**No. of trees**
Training plots	2.15	153.33	1.98	14.18	17.25	14.44	10.34	41.81	2631
Test plots	2.20	142.94	3.80	23.88	18.18	12.89	6.78	34.47	737
All plots	2.15	153.33	2.38	16.30	17.46	14.10	9.56	40.20	3368

Afterwards, each of the training and test plots were duplicated and modified to represent various DBH cutoffs. For each plot (training/test), the TLS plots, which are already pre-processed as shapefiles in the form of polygons with attributes that include area, were subset based on the DBH of trees within each of them. For a DBH cutoff, the initial TLS plot was first duplicated to preserve the original, and all trees less than that cutoff size were removed from the plot. DBH cutoff values used were 5cm, 10cm, 15cm, 20cm and 25cm; 0cm cutoff being the preserved TLS plot. Hence, each training/test plot over various DBH cutoffs typically had variations of the same original plot with larger and

often fewer trees for increasing cutoff sizes in the TLS plots, while the airborne LiDAR plots were unmodified in this scenario.

Genetic Algorithms

Genetic algorithms generally utilizes a population of individuals/chromosomes with encodings of parameters in a binary string (Figure 1), which are often randomly initialized before the start of exploration. For each generation, some of these individuals are randomnly selected for exchange of genes (a combination of bits) in a process called crossover (Goldberg, 1988). This combination process for 2 chromosomes is determined probabilistically, often in proportion to the calculated fitness of individuals. Thus, individuals of higher fitness values are more likely to be paired for crossover. Afterwards, a typically slight probability of mutation can alter the value of one of more bits of an individual. These probably altered individuals proceed to the next generation and the cycle repeats until a reasonable solution is reached.

For a given parameter of range [a, b] and p as its precision, the number of bits for binary representation, n, and decoding is given by:

Precision (p)	=	$(b - a) / (2^n - 1)$
decimal value	=	binary to decimal conversion of bits
Decoded parameter value	=	a + precision × decimal value

Equation 1: Parameter encoding/decoding given range [a, b] occupying n bits in the chromosome

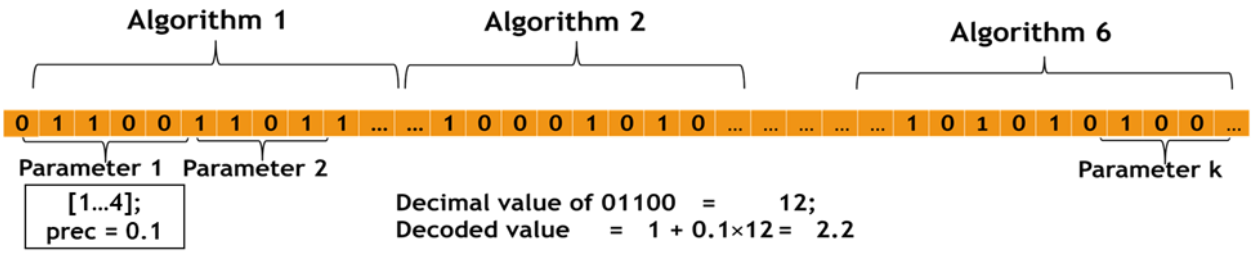


Figure 4: Chromosome illustrating encoding and decoding of parameters.

From Figure 4, "Parameter 1" must be at least 5 bits ($2^n = 32 \geq 31$) for encoding this parameter. To decode the actual value of any encoded ITC's parameter, the binary value is first decoded to base 10. Thereafter, the parameter's value is calculated by adding the minimum acceptable value in its range with the product of this base 10 value and the precision as seen in equation 1.

Individual Tree Crown Recognition Algorithms

Multiple ITC recognition algorithms have been developed, and six (6) of these were used for this study. The choice of these algorithms was based on multiple factors including the fact that they were developed in R, where the genetic algorithms and LiDAR data was being processed. Also, these 6 algorithms were available in R packages and had accessible publications documenting their development and implementation.

These tree crown recognition algorithms include Watershed (Wulder et al. 2000), dalponte2016 ((Dalponte and Coomes, 2016), and silva2016 (Silva et al., 2016), which are all raster-based tree segmentation algorithms available through the lidR package (Roussel et al. 2020; Roussel and Auty, 2022). While li2012 (Li et al., 2012) is a non-raster-based segmentation algorithm also available in lidR. Also, two more segmentation

algorithms – ptrees (Vega et al., 2014) and hamraz2016 (Hamraz et al. 2016) were used and are both available from the lidRplugins package (Roussel, 2022). For the rest of this study, these 6 algorithms would be referred to as Watershed, Dalponte, Silva, Li, Trees and Hamraz. Each of these ITCs have varying numbers of default parameters that are used for tree crown recognition. The values of these discrete parameters were not necessarily always related to actual tree-level attributes, and changes to a single one can significantly affect performance of an ITC.

The parameters to be tuned for each tree segmentation algorithm were identified. To obtain a reasonable range of values and precision/step sizes of tuning for each parameter, several tests were carried out on each tree segmentation algorithm with manual fine-tuning of each parameter's value incremented in fine steps. This was done with a basic performance assessment for the number of identified trees for a known airborne LiDAR plot. This range and step sizes for values were necessary to determine the number of bits for encoding each parameter in a chromosome.

Table 4 shows the individual parameters for each ITC as they were developed, including their identified ranges (both inclusive) and precisions that were fed into the genetic algorithm. The meaning of each parameter can be found in the official R documentation for each ITC. An "algorithm" parameter was used as a pseudo-parameter to select one out of 3 provided algorithms for generating a digital surface model – "p2r" (point-to-raster), "pitfree" (pit-free algorithm) and "dsmtin" (Delaunay triangulation). The kernel size used is the size of the moving kernel for raster-based ITCs – radius for the circular kernels used in Watershed and Dalponte; length for the square kernels used in Silva. Each of

these had its own parameters which were tuned as well before implementing a grid canopy model as the first computational step in the implementation of 3 of the tree segmentation algorithms that utilized such.

Table 4: List of encoded chromosome's parameters (raster-based ITCs make use of pseudo-algorithm parameter)

ITC	Number of parameters	Number of bits	Raster-based
Watershed	7	25	Yes (circular kernel)
Dalponte	10	42	Yes (circular kernel)
Silva	8	33	Yes (square kernel)
Li	4	16	No
Ptrees	2	10	No
Hamraz	9	50	No

Thus, a single chromosome had a binary encoding for all 40 parameters (29 inherently unique with 11 variations for different ITCs) and a total length of 176 bits.

Fitness Function

To evaluate the objective function per chromosome at every generation, an individual is decoded into the values of its various parameters and all 6 ITC recognition algorithms were evaluated using these parameters for a single plot, producing an output LAS file with detected trees at various coordinates identified. This output contains information about the number of uniquely identified trees and their spatial positions.

The objective function is defined here as a measure of deviation from the optimal solution of the GA, and hence the aim is to minimize it. While the fitness function is to be

maximized as it represents a measure of performance of the GA, and is typically derived as an additive or multiplicative inverse of the objective function. Each ITC algorithm identifies and delineates n tree crowns (as a cluster of points with an ID) per plot. Consequently, each segmentation algorithm is evaluated as follows:

Objective function	=	0.5*(Tree count error + Spatial error)
Tree count error	=	Δ_{tree}/ maximum(actual trees, predicted trees)
Spatial error	=	$\overline{\Delta_{\text{dist}}}$/ maximum value of Δ_{dist}

Equation 2: Objective function of genetic algorithm to be minimized

Where Δ_{tree} is the absolute difference in actual and identified trees, and Δ_{dist} is the Euclidean distance from an actual tree to the nearest identified tree by an algorithm – using the rgeos package (Bivand and Rundel, 2021). Δ_{dist} was evaluated for every actual tree on site with the ground truth and averaged to get $\overline{\Delta_{\text{dist}}}$ Details about the actual number of trees and their positions were derived from ground truth data available for each plot in the form of shapefiles using the rgdal package (Bivand et al. 2022).

The multi-objective function in equation 2 was designed with the intention of (as optimally as possible) weighting the impacts of over or under identification of trees, as well as the deviation in coordinates from actual trees on site. This was necessary due to the unequal units of these two error sources, as the mean distance is typically measured in meters and typically had values ranging from a fraction to a few meters. Whereas Δ_{tree} has no unit and can take on any possible non-negative value. Consequently, a perfect tree crown recognition algorithm that identifies the exact number of actual trees (a tree count

error of zero) at their exact locations (a spatial error of zero) would have the least objective function value of zero and hence maximum fitness.

Fitness per algorithm = 1 – Objective Function

Total chromosome fitness = 6 × Fitness per algorithm

Equation 3: Fitness function to be maximized by each chromosome of genetic algorithm.

Fitness sharing was implemented in equation 3 by equally distributing the maximum fitness for each algorithm, and this fitness was carefully chosen to ensure it always remains positive. Moreover, the maximum value of the objective function (corresponding to the minimum fitness) per chromosome is such that it typically never exceeds 1 for even the worst algorithms. Error checking was implemented such that if an unlikely combination of parameters fails to identify a tree, a fitness value of 0.0006 per chromosome (10^{-4} per ITC) is the resulting output.

Fitness sharing also ensured the final solution of the genetic algorithm did not converge to a single tree segmentation algorithm but distributed across all of them, and thus creating multiple optima in the search space. The subsequent effect is an optimization of as many parameters as possible for the different tree segmentation algorithms irrespective of the performance of one or a few of them.

GA Parameters

The entire implementation of the genetic algorithm was done in R using the GA package (Scrucca, 2013). A population size of 200 over at most 500 generations for 176-bit long chromosomes were used. A chromosome with values representative of the defaults

(where available) or sample values of parameters from their documentations for all algorithms was seeded into the GA as the initial point of optimization. These genetic algorithms were run over the training plots. A random seed (77) was used for reproducibility of results.

Each chromosome's objective function and fitness were evaluated using equations 4 and 5 respectively after running all 6 ITC recognition algorithms per fitness evaluation with the decoded parameters. These were then combined into a fitness value with equation 6. Due to the computationally expensive nature of the fitness evaluation – running all 6 tree segmentation algorithms and evaluating the objective functions, parallelization of the population's fitness evaluation was implemented with 100 cores (Scrucca, 2017).

One-point crossover and flip mutation were used with probabilities of 0.95 and 0.05 respectively, while linear rank selection was used to minimize selection pressure, which in turn reduces rapid convergence of solutions unlike what is commonly obtainable in fitness proportional selection (Goldberg and Deb, 1991). To ensure the best individuals are always preserved or only replaced with fitter individuals, elitist selection was utilized with a "keep 6-best" method implemented per generation. The genetic algorithm gets terminated when the fittest individual does not improve after 50 consecutive generations.

Due to the complexity of the search space, specific combinations of parameter values threw errors and error handling was implemented to prevent uninterrupted execution of the genetic algorithm, while the chromosome and the tree segmentation algorithm (just before it is evaluated for fitness) were logged to a file using the logr package (Bosak, 2022).

Accuracy Metrics

The results were evaluated with various metrics, described shortly, to assess and analyze the performance of the genetic algorithm. Also, these metrics were used to evaluate the performance of the genetic algorithm for modified variations of the groundtruth at various DBH cut-offs. While the airborne las files were unmodified, the groundtruth data containing the tree stems were subsampled to contain trees having DBH sizes with minimum threshold values of 5cm, 10cm, 15cm, 20cm and 25cm. For each of these thresholds, the entire genetic algorithm was re-run with the subsampled groundtruth and each of these metrics were re-evaluated on the same subsamples.

Assessment of accuracy and validation of results was done using precision, recall, F-score, root mean squared error (RMSE), percentage bias, and the coefficients of correlation and determination. To calculate the true positives (used in precision and recall), a 1.5m threshold is set as the maximum cut-off distance between a predicted (point cloud cluster) and actual tree stems as being the same tree. For each actual tree on site, since all test plots are non-empty, if there is at least one predicted tree within 1.5m radius of the ground truth, the closest of them is assigned to the actual tree, and this counts as a true positive. While the remaining predicted trees are used for the rest of the iteration. In the event of no predicted tree being within that threshold, this counts as a false negative.

For each iteration, any matched predicted tree is temporarily removed from the rest of the data to avoid multiple counting and thus introducing flaws to the analysis. Consequently, there is one less actual and one less predicted tree on site for each iteration. This iterative

process continues for all in-situ trees as the sum of all matched trees for predictions and reality with the existing ground truth is the total false positives. It is important to note that this threshold is selected with the assumption that tree stems are reasonably expected to be within the canopy due to the typical triangular shape of conifers. Hence, 1.5m was used for this study as the threshold distance for matching tree stems to predicted canopy's centroid (presumably the canopy's center).

For the false positives with each cluster of points identified as belonging to a single tree ID, and therefore an individual tree, the mean of the two-dimensional coordinates is calculated for each of these identified trees by the ITC algorithm. This aggregation of mean coordinates is converted to a simple feature in the same coordinate reference system as the ground truth for comparison. Thereafter, a 1.5m threshold for matching trees is also selected. While there is a predicted tree in the plot, the closest (in spatial proximity) actual tree is matched to it (which is a true positive as previously calculated). However, if there is no actual tree within that threshold, it is considered a false positive. Similar precautions are taken by temporarily removing matched actual and identified trees to avoid double counting, and the process terminates after going through all predicted trees. Thus, the total sum of unmatched trees serves as the false positives. Whereas the false negatives for any plot is the difference between the actual number of trees on site and the true positives.

Landscape Tree Crown Mapping with Optimized Parameters

After evaluating the performance of the ITC models post GA run, the best model among all ITC GA-tuned parameters was chosen for mapping multiple airborne LiDAR plots in

the Lake Tahoe Basin Management Unit. This model selection decision was made based on the F-scores values; with the tree crown recognition algorithm whose F-score was the highest for all DBHs chosen.

Results

GA Fitness Values

The peak fitness values increased up to the 15cm DBH cutoff before plummeting past that cutoff value (Figure 5). Also, all genetic algorithms stopped well before the maximum number of generations set, as seen in Table 5, with convergence typically happening early. The common stopping criterion for all GAs was 50 consecutive maximum fitness values.

Table 5: Maximum number of generations run by genetic algorithms for different DBH cutoffs

DBH minimum cutoff	All sizes	5cm	10cm	15cm	20cm	25cm
Number of generations	109	165	174	216	91	229

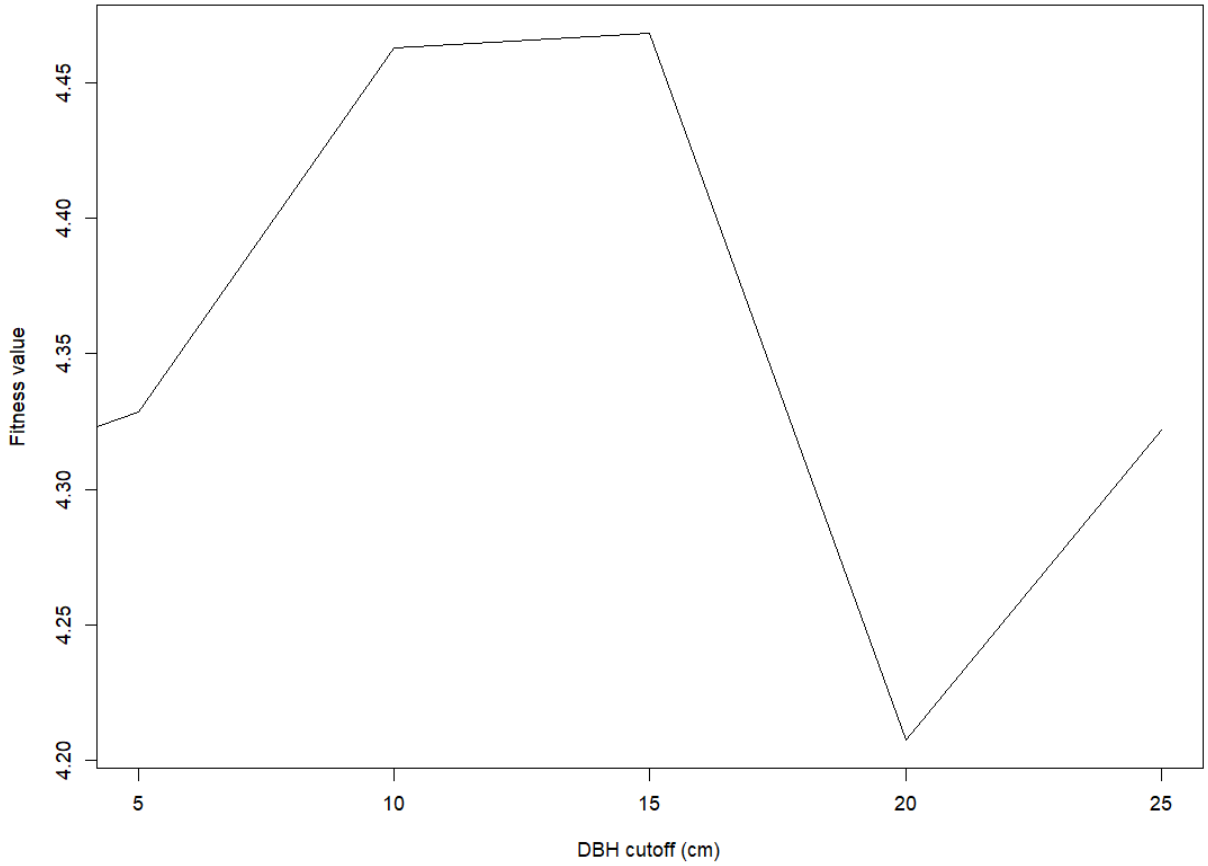


Figure 5: Fitness values of GAs over different DBH cutoffs

Confusion matrix

The precision values from Figure 6 generally decreased for increasing DBH cutoffs, with the genetic algorithm's fine-tuned parameters having higher precision for most DBH size cutoffs across the validation plots. Also, Table 6 shows the 5cm DBH cutoff had the highest precision values for all ITCs, with watershed's default parameters generally being the worst performing at all cutoff values.

However, watershed and Dalponte were the best-performing ITCs for the GA-tuned parameters, outperforming all other ITCs after parameter tuning. Although, at the 15cm DBH cutoff, Silva's default outperformed watershed's GA-optimized parameters and all other ITC default values.

Table 6: Precision values for default and GA-optimized tree crown recognition algorithms (best values in each row are color-coded)

DBH Cutoff (cm)	**Watershed**	**Dalponte**	**Silva**	**Li**	**Ptrees**	**Hamraz**	**GA- optimized (GA-best)**
5	0.1864	0.3955	0.4028	0.4118	0.2676	0.2731	0.4500 (Watershed)
10	0.1533	0.3475	0.3654	0.3665	0.2263	0.2423	0.4048 (Dalponte)
15	0.1205	0.3107	0.3222	0.2941	0.1895	0.2203	0.3161 (Watershed)
20	0.0950	0.2684	0.2692	0.2353	0.1460	0.1718	0.2925 (Dalponte)
25	0.0817	0.2373	0.2338	0.1946	0.1237	0.1454	0.2602 (Dalponte)

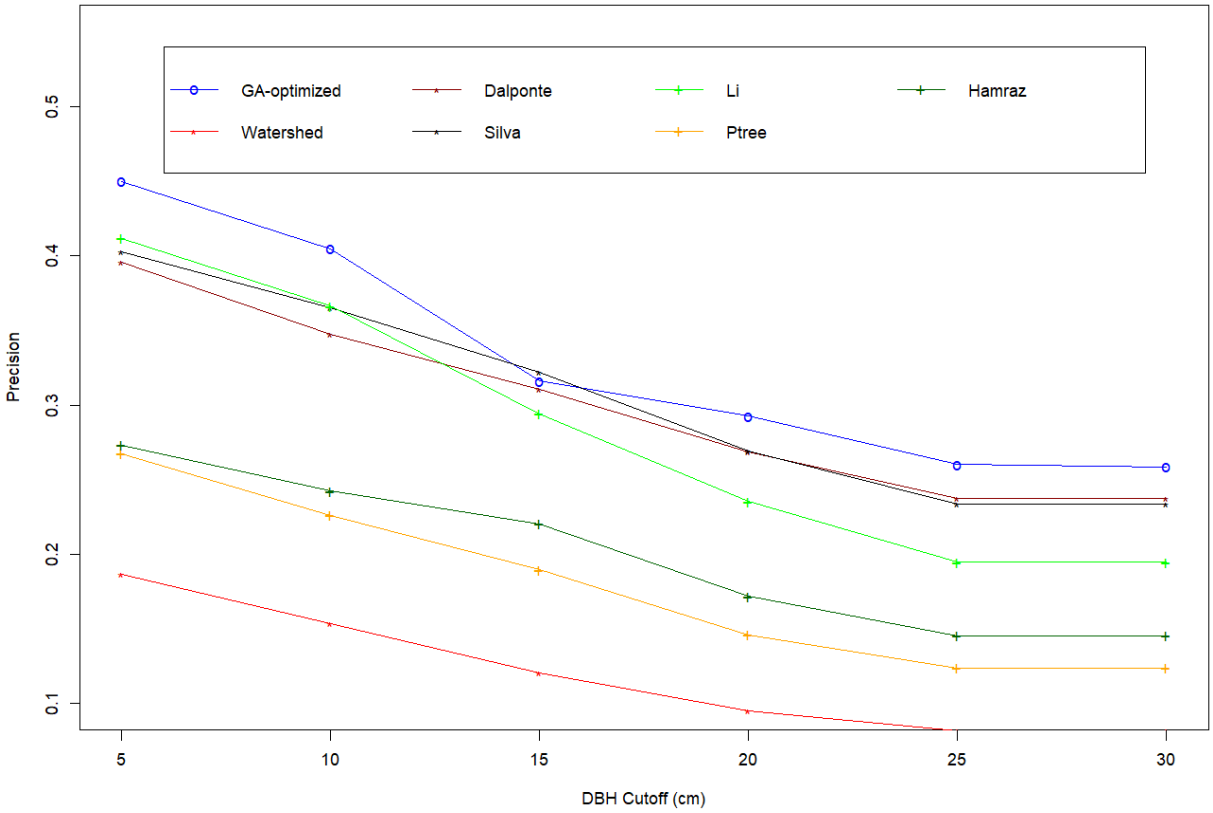


Figure 6: Plot of Precision versus DBH cutoffs

Either Li or Hamraz had the worst recall across all DBH cutoffs as seen in Table 7. Also, either Silva or Ptrees among the ITCs had the best recall values across all DBH cutoffs for GA-optimized parameters. Watershed's default parameters outperformed all other ITC defaults and even the GA-tuned predictions in the recall values. Whereas the trend of recall values from Figure 7 is the reverse of that for precision, as it generally increases for higher DBH cutoffs.

Table 7: Recall values for default and GA-optimized tree crown recognition algorithms (best values in each row are color-coded)

DBH Cutoff (cm)	Watershed	Dalponte	Silva	Li	Ptrees	Hamraz	GA- optimized (GA-best)
5	0.7306	0.1975	0.2891	0.1283	0.3385	0.0874	0.3385 (Ptrees)
10	0.7992	0.2308	0.3490	0.1520	0.3809	0.1032	0.3058 (Silva)
15	0.8396	0.2757	0.4110	0.1629	0.4261	0.1253	0.3534 (Silva)
20	0.8684	0.3125	0.4507	0.1711	0.4309	0.1283	0.6711 (Silva)
25	0.8937	0.3307	0.4685	0.1693	0.4370	0.1299	0.4370 (Ptrees)

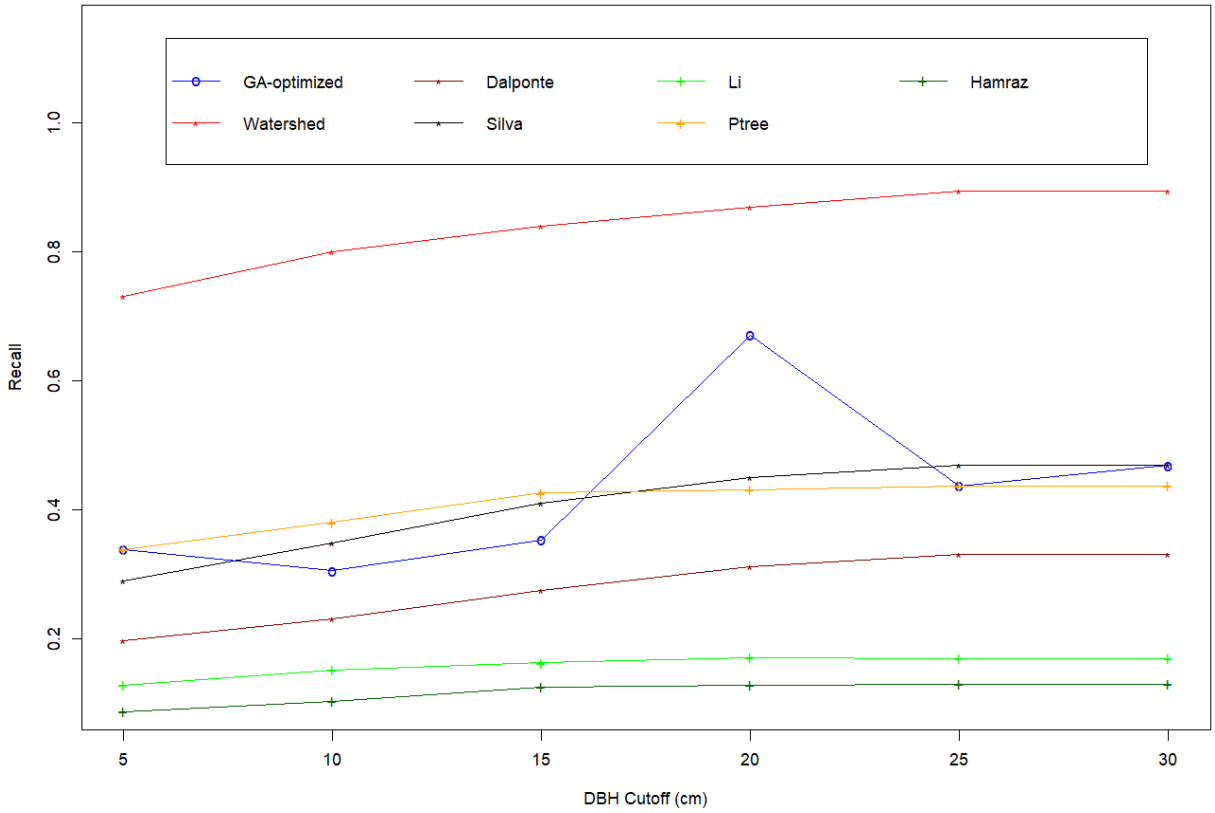


Figure 7: Plot of Recall versus DBH cutoffs

The F-scores in Figure 8 do not show any specific trend across DBH values. Silva's default had the highest F-score values across all DBH cutoffs for all ITCs from Table 8. Quite interestingly, Silva's default outperformed the GA-tuned parameters for all but one DBH value, 20cm, which was led by same Silva's GA-tuned parameters. Once again, either Li or Hamraz generally had the least F-scores across DBH cutoffs.

Table 8: F-scores for default and GA-optimized tree crown recognition algorithms (best values in each row are color-coded)

DBH Cutoff (cm)	Watershed	Dalponte	Silva	Li	Ptrees	Hamraz	GA- optimized (GA-best)
5	0.2970	0.2634	0.3366	0.1957	0.2989	0.1325	0.2989 (Ptrees)
10	0.2572	0.2773	0.3570	0.2149	0.2839	0.1447	0.3267 (Silva)
15	0.2108	0.2922	0.3612	0.2097	0.2623	0.1597	0.3264 (Silva)
20	0.1713	0.2888	0.3370	0.1981	0.2182	0.1469	0.3585 (Silva)
25	0.1497	0.2763	0.3119	0.1811	0.1929	0.1372	0.2897 (Dalponte)

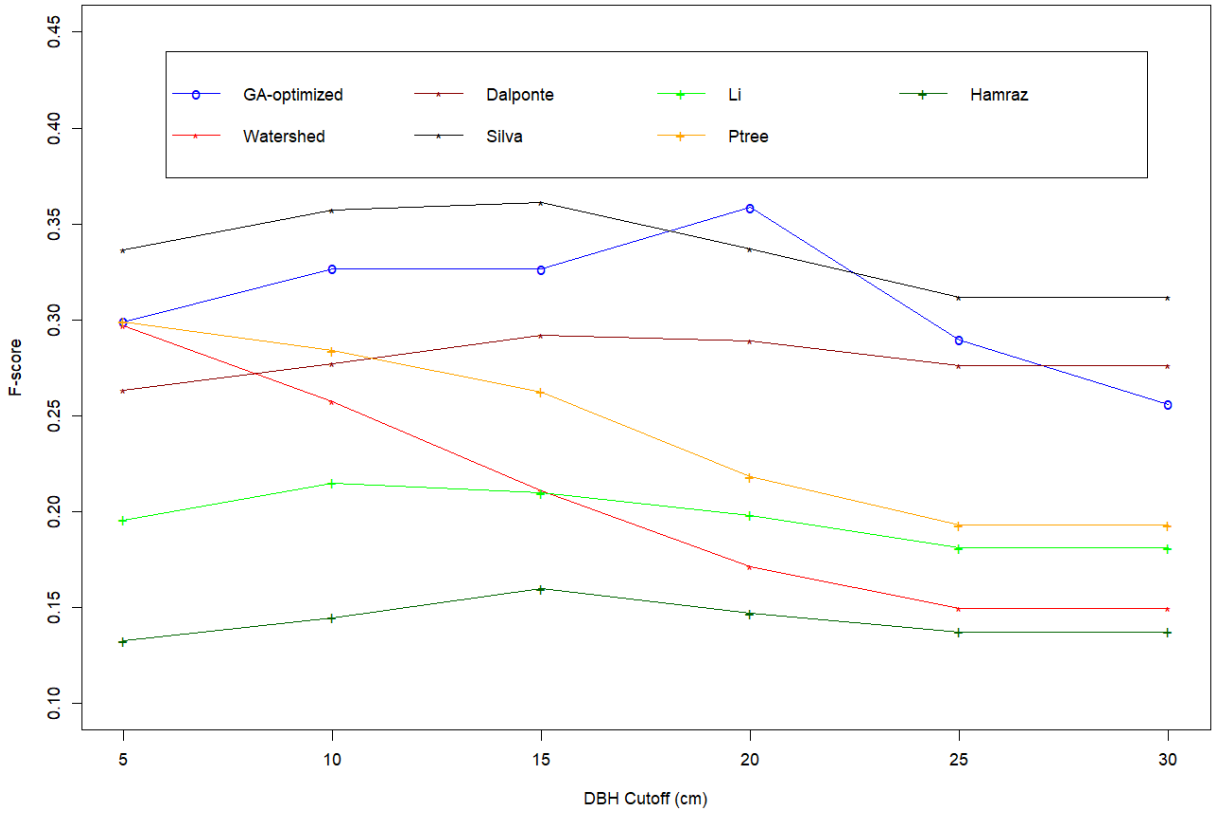
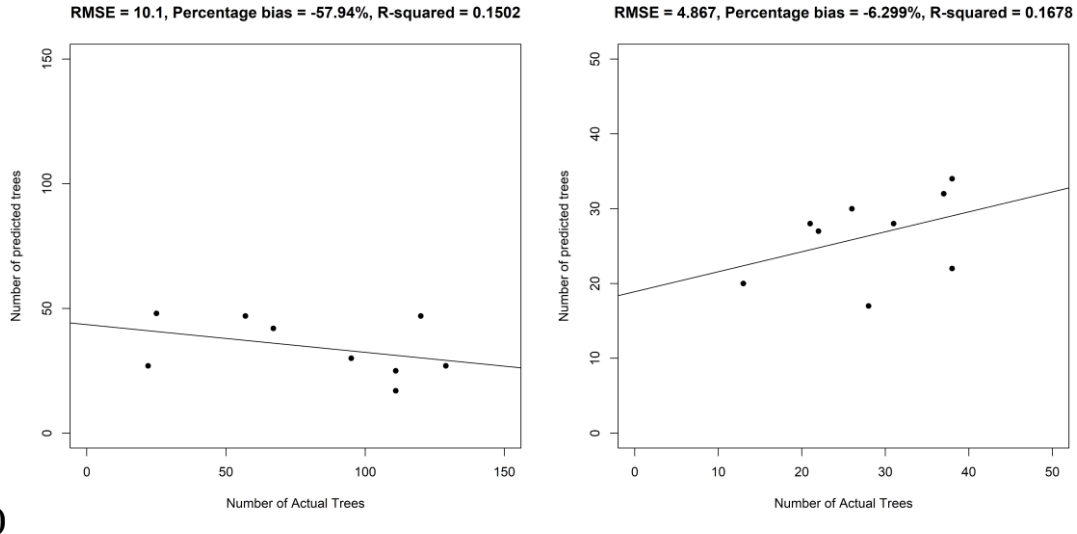


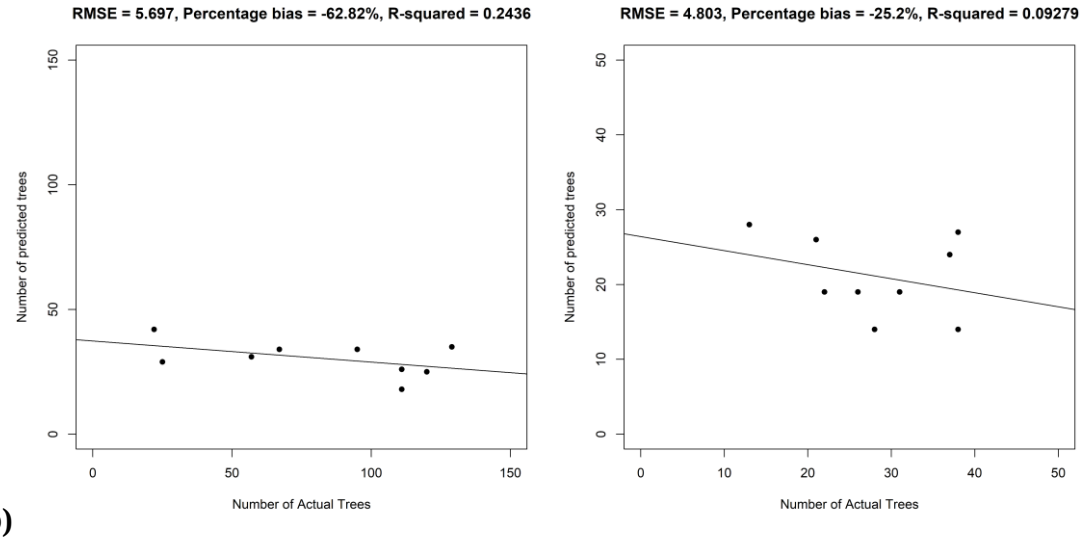
Figure 8: Plot of F-scores versus DBH cutoffs

Predicted Trees

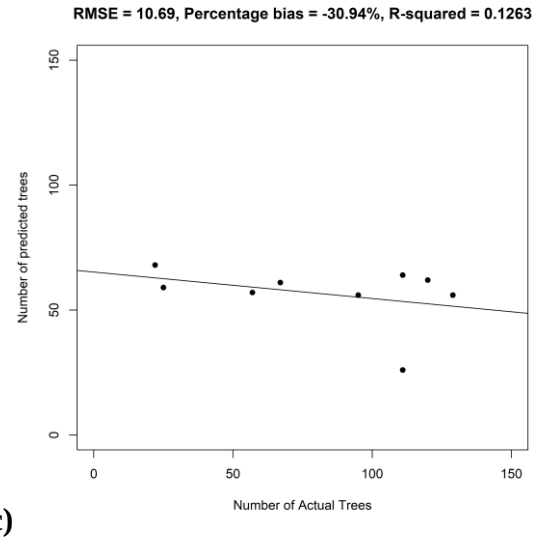
The number of predicted trees across the test plots for all ITCs after GA-tuning of parameters is shown in Figure 9. This prediction on the airborne LiDAR was first compared with tree stems of at least 5cm DBH, and afterwards tree stems of at least 25cm DBH to compare trends of predictions for all trees versus large trees. There is generally a high negative percentage bias for trees with at least 5cm DBH, while this percentage bias was less negative when compared with trees of at least 25cm DBH. In a few cases for the latter, the percentage bias was a large positive value.

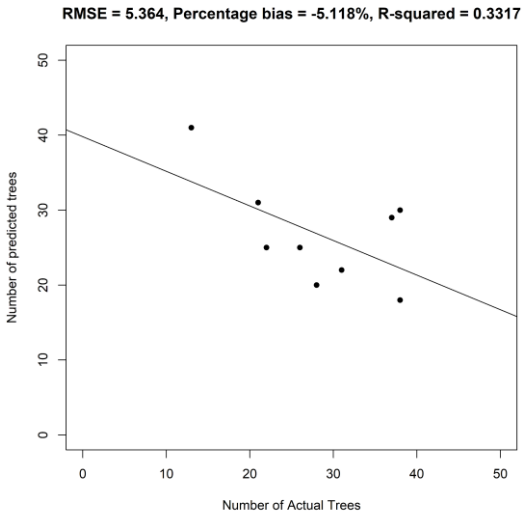

RMSE = 10.1, Percentage bias = -57.94%, R-squared = 0.1502
Number of predicted trees
Number of Actual Trees
0
50
100
150
RMSE = 4.867, Percentage bias = -6.299%, R-squared = 0.1678
Number of predicted trees
Number of Actual Trees
0
10
20
30
40
50

(a)

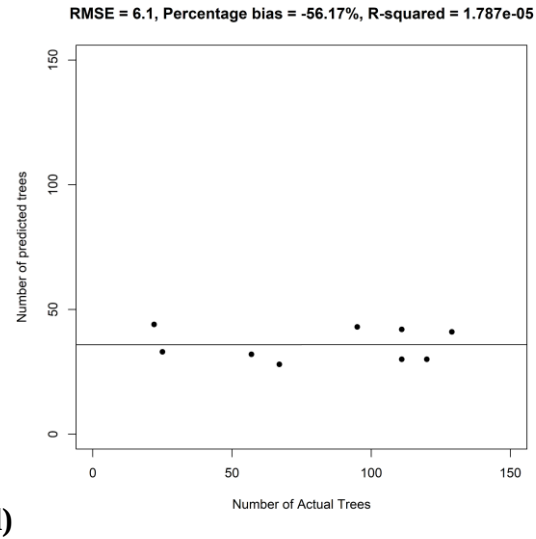

RMSE = 5.697, Percentage bias = -62.82%, R-squared = 0.2436
Number of predicted trees
Number of Actual Trees
0
50
100
150
RMSE = 4.803, Percentage bias = -25.2%, R-squared = 0.09279
Number of predicted trees
Number of Actual Trees
0
10
20
30
40
50

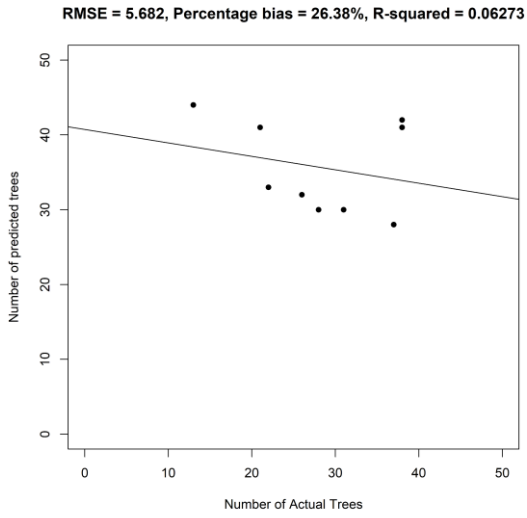
(b)


RMSE = 10.69, Percentage bias = -30.94%, R-squared = 0.1263
Number of predicted trees
Number of Actual Trees
0
50
100
150

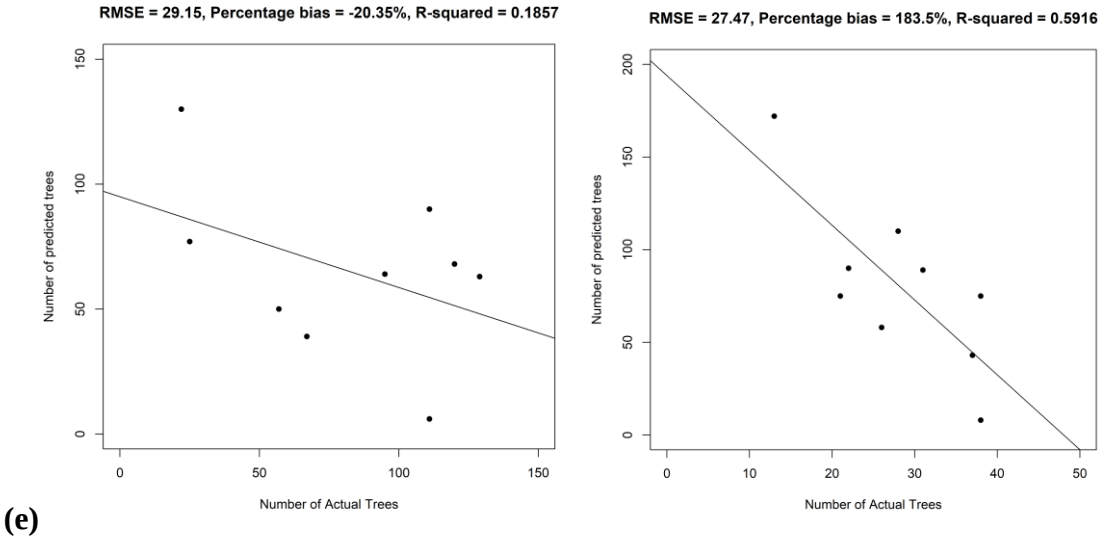

RMSE = 5.364, Percentage bias = -5.118%, R-squared = 0.3317
Number of predicted trees
Number of Actual Trees
0
10
20
30
40
50

(c)

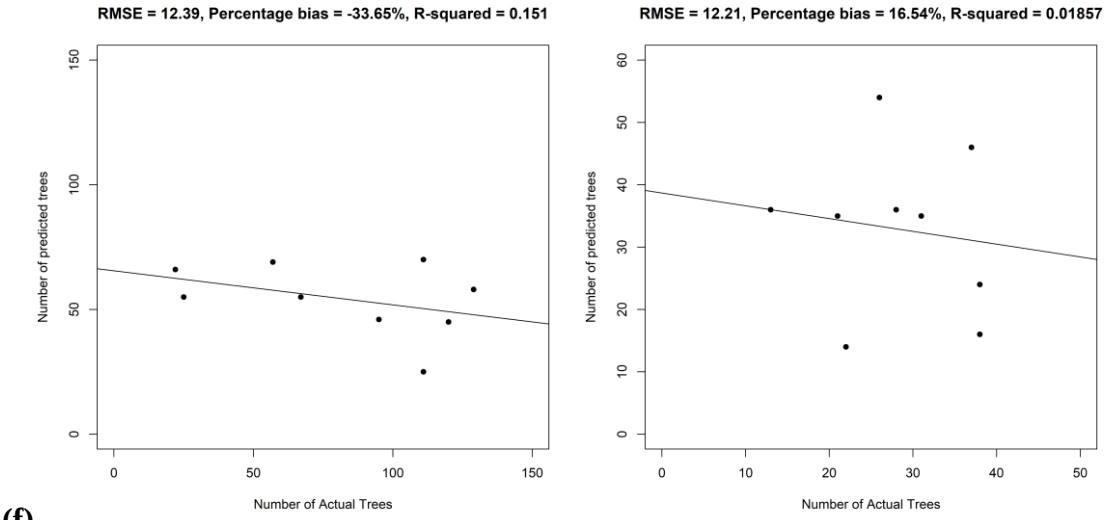

RMSE = 6.1, Percentage bias = -56.17%, R-squared = 1.787e-05
Number of predicted trees
Number of Actual Trees
0
50
100
150


RMSE = 5.682, Percentage bias = 26.38%, R-squared = 0.06273
Number of predicted trees
Number of Actual Trees
0
10
20
30
40
50

(d)



(e)



(f)

Figure 9: Predicting trees with GA-tuned parameters validated with tree stems of minimum DBH: 5cm (Left); 25cm (Right)
(a) Watershed (b) Dalponte (c) Silva (d) Li (e) Ptrees (f) Hamraz

Tree Crown Maps at Landscape Level

Tree crown maps of an LTBMU plot at a landscape scale were produced with all ITC models with the GA-optimized parameters, as shown in Figure 11. This plot measures 275m × 265m, and a tree crown prediction comparison is shown for them.

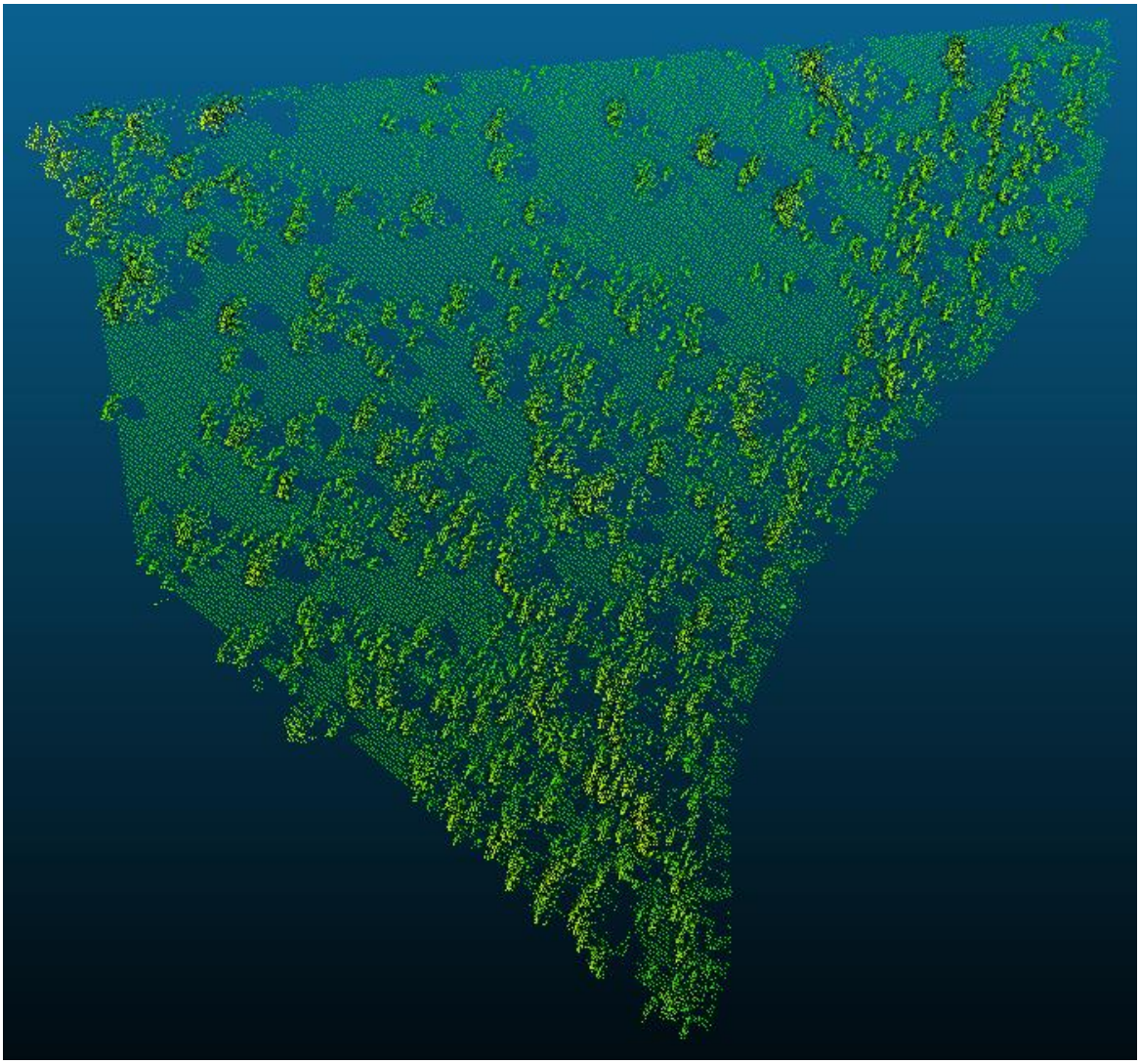
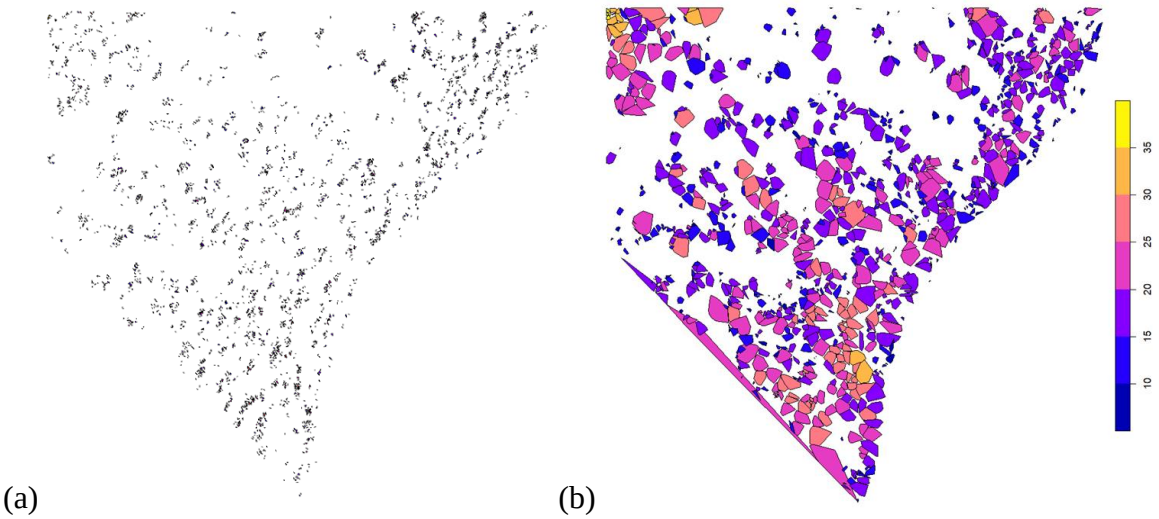
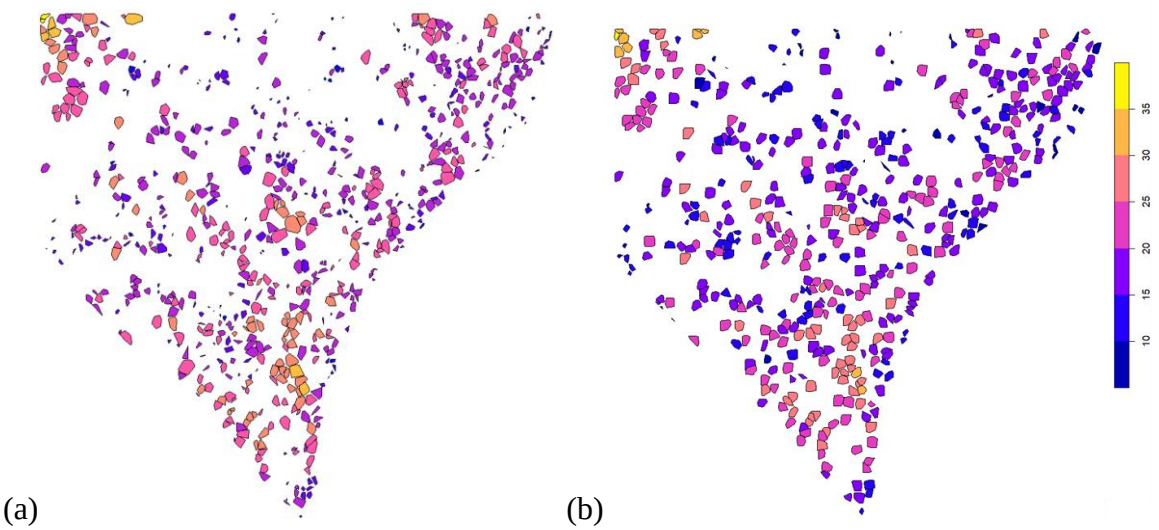


Figure 10: LTBMU airborne LiDAR plot (275m × 265m)



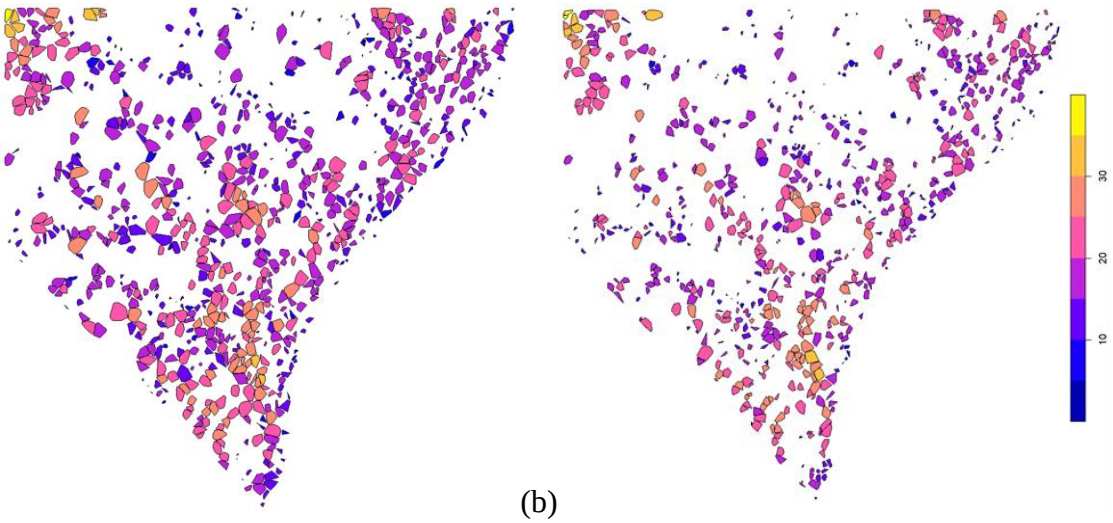
(a) (b)

Figure 11: Watershed's tree crown map
(a) Default parameters (18,027 trees) (b) GA-optimized parameters (1,107 trees)



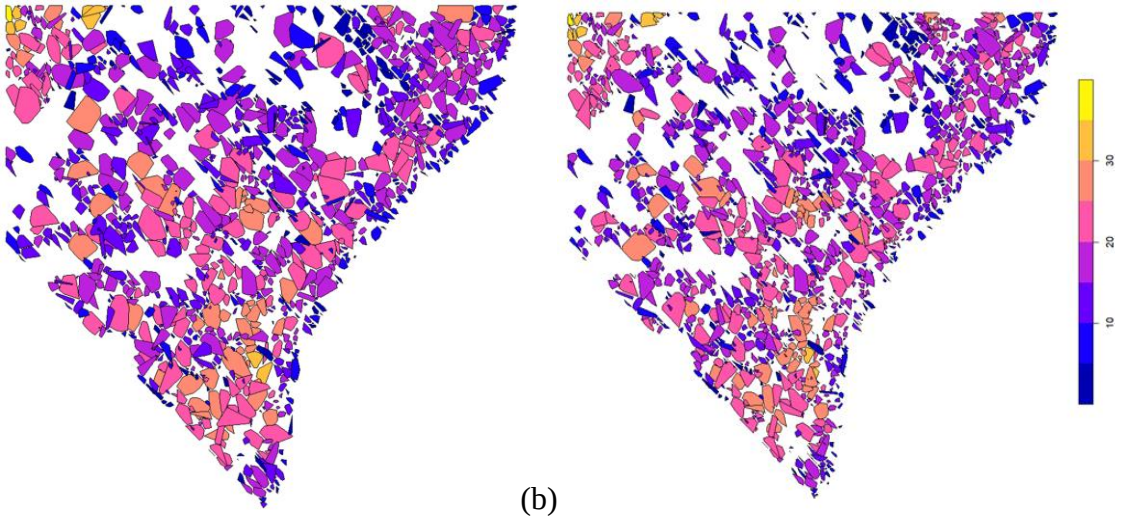
(a) (b)

Figure 12: Dalponte's tree crown map
(a) Default parameters (675 trees) (b) GA-optimized parameters (507 trees)



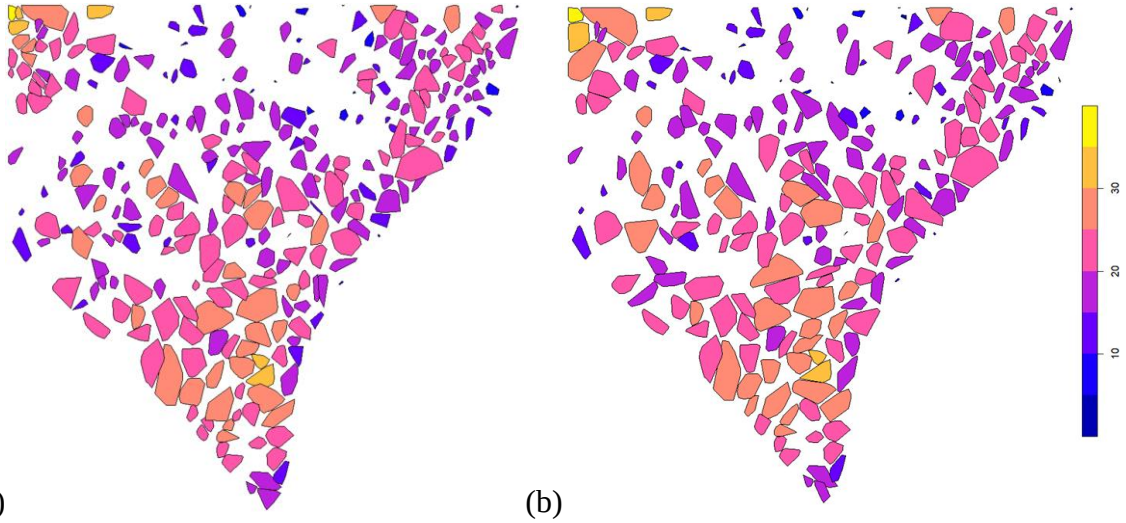
(a) (b)

Figure 13: Silva's tree crown map
(a) Default parameters (871 trees) (b) GA-optimized parameters (782 trees)



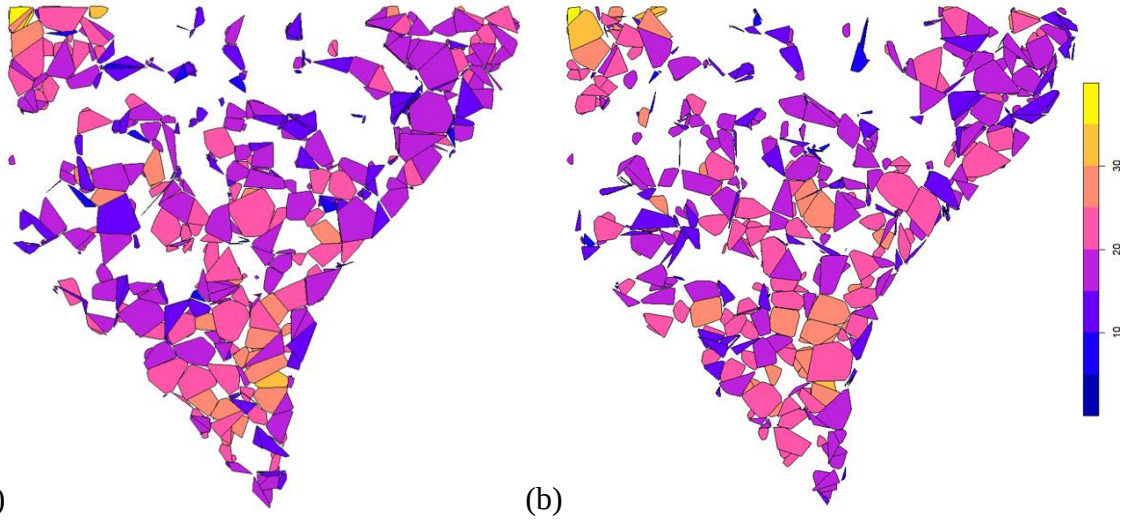
(a) (b)

Figure 14: Li's tree crown map
(a) Default parameters (1,177 trees) (b) GA-optimized parameters (1,507 trees)



(a) (b)

Figure 15: Ptrees' tree crown map
(a) Default parameters (311 trees) (b) GA-optimized parameters (243 trees)



(a) (b)

Figure 16: Hamraz's tree crown map
(a) Default parameters (586 trees) (b) GA-optimized parameters (599 trees)

Discussion and Conclusion

GA-optimization of parameters did show we can significantly improve the performance of most ITCs for tree crown mapping as the results have shown. The fitness function for the GA played a significant role in achieving these results, as its dual-optimization purpose for tree count and spatial locations sought to strike a balance between predicting the actual number of trees and making these predictions at the right locations. Evaluations of GAs were not based on the same fitness function, but rather on F-scores as it is important to objectively evaluate on an independent but relevant metric that is intuitive to the remote sensing and ecological implications.

Consequently, the fitness functions in Figure 5 tells very little about the actual performance, as it is a purely mathematical metric meant to numerically evaluate each chromosome. Although it is developed with the actual problem statement in mind, the remote sensing problem and subsequent application is less mathematical than ecological. This is because a plot with 70 trees could, in theory, be more closely satisfied by a GA-solution that predicts a hundred trees where at least one predicted tree matches the location of an actual one, than a more realistic prediction of 56 trees with a slight offset from actual ones. This is important to note in the context of the typically larger trees that ITCs generally detect and are often of more practical importance. The F-scores are meant to more closely represent how a human would think about how a tree crown prediction algorithm should match reality in forests, based on how true positives, false positives and false negatives were evaluated.

The general decreasing trend of precision values as DBH sizes increase (Figure 6) suggests either an increase in the number of false positives, or a decrease in the number of true positives, or both. The combination is more likely as the number of true positives is expected to decline due to the presence of less actual trees on site since smaller tree sizes were excluded from the groundtruth in larger DBH cutoffs when evaluating precision.

Also, the number of false negatives would increase because the same previously detected small trees from the airborne LiDAR, in the excluded DBH size would still be predicted by the GA-tuned parameters of the tree crown recognition algorithms. But these small trees would appear as false negatives due to their exclusion from the ground truth. The increase in false negatives is expected to be significantly higher than the decrease in true positives for a declining trend to be the outcome in the equation defining precision.

In most cases, dalponte and watershed when fine-tuned by the genetic algorithm had the highest precision across various DBH values from Table 6, and experienced the largest increase in this metric. This improvement is significant considering their default parameters had precision values always strictly less than the GA-optimized values. Ptrees, Hamraz and Watershed had the worst precision values when tree crown predictions are made with their default parameters, suggesting a tendency to predict more non-existent trees than others.

However, recalls had a different trend with a general to improve for large DBH cutoff values (Figure 7). Consequently, it's either there is an increase in true positives or a decrease in false negatives or both. The previous analysis of precision already suggests an increase in true positives for higher DBH cutoffs is quite unlikely. Hence, the most

probable reason for this increasing trend of recall is lower false negatives. A simple technical analysis of the datasets for various DBH cutoffs shows that since there are fewer actual trees in the ground truth as the cutoff size increases, due to exclusion of smaller trees. Consequently, the number of undetected actual trees would likely reduce significantly. This likely implication is further strengthened by the fact that the remaining trees are of much larger sizes and easy to detect, since larger DBH generally implies larger trees.

It is also important to note that watershed's default parameters outperformed that of all other ITCs and even the GA-tuned predictions in recall. While this might seem unexpected, watershed uses a moving window on a canopy height model to detect trees based on peak irradiance is often prone to commission errors at fine spatial resolutions or for trees at close proximity (Wulder, 2000). Lt was developed on forest stands with median DBH of 24cm, which is much larger than the one used for this study. Moreover, the commission errors of the watershed algorithm for a fixed window size (which is used by default) tends to predict non-existent trees, which can eventually match some actual trees on site, even while lowering the precision. It's default parameters grossly overestimated trees on site, with micro tree crown shapes (Figure 11) as predictions. However, its crown mapping ability significantly improved after GA-tuning even though the tree crown shapes especially along edges were rather sharp.

Beyond watershed, the GA-tuned parameters performed just as well as the best of other ITC default parameters for recall, with a sharp peak at 20cm DBH cutoff. This is probably the point at which the false negatives were least due to detection of almost all

actual trees relative to the true positives, such that the equation for recall had a much higher value than other DBH values. Li and Hamraz had the worst recall across all DBH cutoffs with very little tendency to improve which might either suggest a poor ability to predict trees at actual tree locations. Hence more false negatives, or a very conservative prediction ability in general, and less true positives.

Silva and Ptrees led the recall values for GA-optimized parameters from Table 7 and generally experienced either none or a negative increase in the performance of their default parameters. Hence, parameter tuning by GAs probably did not have a major effect on the precision of ITCs in general. However, Silva and Ptree's default parameters seemed to perform at least as well as the GA-optimized parameters at all but the peak DBH cutoff values.

The F-scores in Figure 8, which represent the harmonic mean of precision and recall do not show any specific trend across DBH values which isn't surprising considering the opposing trends shown by its components. Silva generally had the highest F-score values for various DBH cutoffs and outperformed the GA-tuned parameters for all but one DBH value, 20cm, which was the same as the peak value of the GA-tuned recall parameters. This probably implies Silva's default parameters were expertly chosen and tuned prior to publishing such, considering it also generally performed quite close to the GA-tuned parameters for precision and recall as well. Also, its tree crown predictions (Figure 13) seemed to be the most realistic when compared to the landscape plot (Figure 10) it was applied to.

Once again, Li and Hamraz generally had the least F-scores across DBH cutoffs, with watershed experiencing a steep decline in F-score values at higher DBH cutoffs. This implication is critical when using such ITC algorithms for tree classification from airborne LiDAR, as it suggests their default parameters would need a lot more fine-tuning before use in large-scale practical applications.

However, the GA-optimized parameters had one of the best F-score values and Silva led in most DBH cutoffs among all ITC algorithms from Table 8. It is noteworthy to observe that Silva's default parameters outperformed the GA-tuned parameters for all but a single DBH cutoff, the 20cm cutoff, whose faults was outperformed by the GA-optimized Silva's parameters. An explanation for this would be due to the fitness sharing concept implemented in the genetic algorithm since all ITC algorithms and their parameters were encoded in a single chromosome and jointly used in the evaluation of the fitness of each individual. Thus, highly fit individuals in the population might have a more balanced spread of fitness values across each ITC rather than optimizing for any single one. Consequently, a decline in fitness of a single ITC's parameters in relation to its default can be counterbalanced by significant gains in fitnesses of other ITCs.

The results from Figure 9 show a very weak linear relationship between predicted and actual trees for the ITCs, as seen by their R^2 values, with rather high RMSE values for watershed, ptrees and hamraz. The overall trend for trees of at least 5cm DBH is a gross underestimation tendency, indicated by the typically high negative percentage bias each has. When the same tree crown predictions were made on trees of at least 25cm DBH cutoff, the percentage bias values all tended towards a positive increment, suggesting the

ITCs are likely capable of better detecting large trees compared to the mix of trees of varying sizes. Spatially continuous canopy layers in the plots can also lead to the classification of multiple closeby trees as a single tree.

The 3 raster-based ITCs, in particular, still had negative percentage bias values much closer to zero for these large trees. While the non-raster based ITCs had their percentage bias significantly change to large positive values, indicating significant overestimation tendencies. Ptrees was the most drastic, with a general likelihood of overpredicting the number of large trees by over 160%. This might be likely due to oversplitting of large, overstory trees with spread out branches, while losing most of the small trees present. This loss of small trees becomes evident it was applied to a landscape level plot (Figure 15); it seemed to predict much large tree crowns than the reality. This result seems to be suggested in the development of the algorithm where dominant trees had a 96% detection rate compared to a 44% for dominated trees (Vega et al., 2014).

From the literature on the ITCs used, Silva (Silva et al., 2016) and Watershed (Wulder et al., 2000) were developed on trees with minimum DBHs of 9.5cm and 9cm respectively. Also, the density of the plots used for most ITCs were significantly less than this study, with Silva having an average density of 290 trees per hectare and watershed having an average maximum density of about 200 trees per hectare. These values compared to the 812 and 910 trees per hectare for the training and test plots of this study respectively imply tree density had a significant impact on the performance of the ITCs.

Hamraz, which had the worst recalls and very low precisions for all DBH cutoffs, had a tendency to predict large crown shapes (Figure 16) like ptrees, with GA-tuning having

little effect on the number of identified trees on the landscape plot. This result is supported in Hamraz et al. (2016), which had better recognition for overstory/dominant trees. While dalponte, alongside Silva, closely matched the actual tree shapes (Figure 12) with fewer but larger tree crowns than Silva, while givinga an apparent indication of more sparsity than reality. This is probably due to overstory detection as well, with this ITC model known to have a very poor detection rate for small trees (Dalponte et al., 2016).

Although Li's precision was among the highest for ITC defaults, its tree crown predictions still seemed poor in shapes and sizes with lots of predicted trees (Figure 14), and GA-tuning of its parameters having the effect of reducing predicted crown sizes. Since there is no validation data for the landscape level plot, we can make no inferences about the predicted number of trees. But it is important to highlight the high correlation between number of predicted and actual trees reported in Li et al. (2012), which could be a plausible reason for the high number of predicted trees on the plot.

Genetic algorithms have shown to be highly effective in simultaneously tuning the parameters of various tree segmentation algorithms for optimal performance. Despite the computational cost of the fitness function, GA proves to be a preferable method to the manual tuning of parameters, of which all the ITCs had multiple parameters with varying ranges and precisions. However, the interpretation of the fitness values of the segmentation algorithms in the context of the problem statement is not as direct as the fitnesses imply. Careful considerations of the application of the results and desired

outcomes precede the fitness statistics. Visualization of results often helps to put into context the apparent performance of the algorithms on the domain problem.

The various tree crown recognition algorithms generally underestimated the actual number of trees on each plot by a significant margin. This can be possibly addressed by adjusting the relative weights of penalties applied to displacement and tree count errors. Perhaps a manual or some optimization-based approach for tuning the weight of the penalty applied to the tree count error in the objective function would have helped. Also, a new fitness function can be defined that addresses the combined issues of precise tree counting and positioning. Future studies could consider utilizing a statistic like the kappa coefficient which would be based on the number of identified trees (by each segmentation algorithm) within a set threshold distance from actual trees in the ground truth, and the equivalent for "absence trees" which are randomly generated on the site within certain distance limits from the actual trees present. Also, intersection over union for tree canopy predictions would be a useful way of evaluating future studies. Other fitness functions can be designed depending on the desired objective of the tree segmentation problem or pareto-optimality integrated can be used.

The issue of parameter dependence of tree segmentation algorithms also needs to be addressed for similar problems in the future. Duplicates of the same parameter utilized by various algorithms can be encoded in the chromosome, and each independently optimized and utilized by each tree segmentation algorithm as they may be implemented in different ways. The consequence of this would be longer chromosomes and increased computational complexity. But as this study showed, the chromosome length has very

little bearing on the time computational complexity since each parameter is decoded quickly from its binary encoding to its precise value, before being used in the computationally expensive fitness function that implements all 6 segmentation algorithms. Moreover, the precision and range of certain parameters can be refined to increase the search space and possibly improve the performance, which has the same effect of increasing the bit size of those parameters.

While this study was mostly evaluated in the context of the existing ITC algorithms alone, a different perspective on the strengths and limitations of a GA-approach could have been thrown by comparing it to other optimization approaches including machine learning techniques (Lin and Zhang, 2020), hill climbers and greedy algorithms. We can only confidently assert that a GA-approach is more computationally efficient than an exhaustive search for the number of bits which has a search space of 2^{176}, which can take an unreasonable amount of time by even the most computationally efficient computing systems. Other optimization approaches would have offered insight into the relative time complexity and optimality of parameter convergence for this study.

Existing ITC models used for this study have no filter for assessing tree sizes beyond height thresholds and the size of moving windows in relation to the resolution of the canopy height models for the raster-based ITCs. Properties such as canopy area are not accounted for, and large branched trees might be oversplit, as very small cluster of point cloud indicative of canopies are not overlooked as independent trees. It might also be worth investigating properties of the forested regions as there might be an almost always direct correlation between the values of certain parameters and some of the characteristics

of forest stands. Knowledge of this might improve the computational expense of the GAs as such values can be seeded as part of the initial population or not encoded in the chromosome for tuning, and with increased computational speed be integrated into real-time applications.

Nonetheless, GAs have shown wide applications in various fields and its optimization power shown in this study is a property that could be greatly utilized in the natural resource and ecological discipline, once the problem of interest can be reduced to an optimization function. Hence, GAs can be applied to any ecological problem that can be modelled mathematically and requires selection of attributes and/or scenarios among a range of possibilities, with each having different implications that can be modelled. And while this study did improve on existing ITC's default parameters, it is still limited by the inherent limitations of the original ITCs on which the GAs were optimizing. And hence, genetic algorithms are not algorithms for identifying trees themselves. Future studies would address the development of new tree recognition algorithm that identifies trees in various forest stand structures, canopy layers and densities, which seemed to be a general limitation in the performance of the existing models. Integrating this development with machine learning techniques, especially computer vision, might improve the performance of tree crown mapping. It would still be worthwhile to tune the parameters of newly developed models with genetic algorithms as this study showed its capacity to improve on default values.

References

Abutaleb, K., Newete, S. W., Mangwanya, S., Adam, E., & Byrne, M. J. (2021). Mapping eucalypts trees using high resolution multispectral images: A study comparing WorldView 2 vs. SPOT 7. The Egyptian Journal of Remote Sensing and Space Science, 24(3), 333-342.

Ajayi, O. G., & Ojima, A. (2022). Performance evaluation of selected cloud occlusion removal algorithms on remote sensing imagery. Remote Sensing Applications: Society and Environment, 25, 100700.

Alonzo, M., McFadden, J. P., Nowak, D. J., & Roberts, D. A. (2016). Mapping urban forest structure and function using hyperspectral imagery and lidar data. Urban Forestry & Urban Greening, 17, 135-147.

Ayrey, E., Fraver, S., Kershaw Jr, J. A., Kenefic, L. S., Hayes, D., Weiskittel, A. R., & Roth, B. E. (2017). Layer stacking: a novel algorithm for individual forest tree segmentation from LiDAR point clouds. Canadian Journal of Remote Sensing, 43(1), 16-27.

Bivand R, Keitt T, Rowlingson B (2022). _rgdal: Bindings for the 'Geospatial' Data Abstraction Library_. R package version 1.6-2, https://CRAN.R-project.org/package=rgdal

Bivand R, Rundel C (2021). _rgeos: Interface to Geometry Engine - Open Source ('GEOS')_. R package version 0.5-9, https://CRAN.R-project.org/package=rgeos

Bosak D (2022). _logr: Creates Log Files_. R package version 1.3.2, https://CRAN.R-project.org/package=logr

Brolly, G., & Kiraly, G. (2009). Algorithms for stem mapping by means of terrestrial laser scanning. Acta Silvatica et Lignaria Hungarica, 5, 119-130.

Bunting, P., Lucas, R. M., Jones, K., & Bean, A. R. (2010). Characterisation and mapping of forest communities by clustering individual tree crowns. Remote Sensing of Environment, 114(11), 2536-2547.

Chang, L. C., Chang, F. J., Wang, K. W., & Dai, S. Y. (2010). Constrained genetic algorithms for optimizing multi-use reservoir operation. Journal of Hydrology, 390(1-2), 66-74.

Cheng, D., Li, S., Zhang, H., Xia, F., & Zhang, Y. (2021). Why dataset properties bound the scalability of parallel machine learning training algorithms. IEEE Transactions on Parallel and Distributed Systems, 32(7), 1702-1712.

Culvenor, D. S. (2002). TIDA: an algorithm for the delineation of tree crowns in high spatial resolution remotely sensed imagery. Computers & Geosciences, 28(1), 33-44.

Dalponte, M., & Coomes, D. A. (2016). Tree-centric mapping of forest carbon density from airborne laser scanning and hyperspectral data. Methods in ecology and evolution, 7(10), 1236-1245.

Damanik, I. S., Windarto, A. P., Wanto, A., Poningsih, Andani, S. R., & Saputra, W. (2019, August). Decision tree optimization in C4. 5 algorithm using genetic algorithm. In Journal of Physics: Conference Series (Vol. 1255, No. 1, p. 012012). IOP Publishing.

Davis, Richard C. (September 29, 2005), National Forests of the United States (PDF).

Di Gesù, V., & Bosco, G. L. (2005, September). Image segmentation based on genetic algorithms combination. In International Conference on Image Analysis and Processing (pp. 352-359). Springer, Berlin, Heidelberg.

Dong, P., & Chen, Q. (2017). LiDAR remote sensing and applications. CRC Press.

Draper, F. C., Asner, G. P., Honorio Coronado, E. N., Baker, T. R., García-Villacorta, R., Pitman, N. C., ... & Baraloto, C. (2019). Dominant tree species drive beta diversity patterns in western Amazonia. Ecology, 100(4), e02636.

Duncanson, L. I., Cook, B. D., Hurtt, G. C., & Dubayah, R. O. (2014). An efficient, multi-layered crown delineation algorithm for mapping individual tree structure across multiple ecosystems. Remote Sensing of Environment, 154, 378-386.

Enright, N. J., & Hartshorn, G. S. (1981). The demography of tree species in undisturbed tropical rainforest. Age and growth rate of tropical trees, 107-119.

Esri Inc. (2020). ArcMap [Software]. Version 10.8.1. Redlands, CA: Environmental Systems Research Institute.

Eysn, L., Hollaus, M., Lindberg, E., Berger, F., Monnet, J. M., Dalponte, M., … Pfeifer, N. (2015). A benchmark of lidar-based single tree detection methods using heterogeneous forest data from the Alpine Space. Forests, 6(5), 1721–1747. https://doi.org/10.3390/f6051721

Fassnacht, F. E., Latifi, H., Ghosh, A., Joshi, P. K., & Koch, B. (2014). Assessing the potential of hyperspectral imagery to map bark beetle-induced tree mortality. Remote Sensing of Environment, 140, 533-548.

Ferraz, A., Bretar, F., Jacquemoud, S., Gonçalves, G., Pereira, L., Tomé, M., & Soares, P. (2012). 3-D mapping of a multi-layered Mediterranean forest using ALS data. Remote Sensing of Environment, 121, 210-223.

Forbes, B., Reilly, S., Clark, M., Ferrell, R., Kelly, A., Krause, P., ... & Bentley, L. P. (2022). Comparing Remote Sensing and Field-Based Approaches to Estimate Ladder Fuels and Predict Wildfire Burn Severity. Frontiers in Forests and Global Change, 5.

Girardeau-Montaut, D. (2020). Cloudcompare

Goldberg, D. E., & Deb, K. (1991). A comparative analysis of selection schemes used in genetic algorithms. In Foundations of genetic algorithms (Vol. 1, pp. 69-93). Elsevier.

Goldberg, D.E. (1988). Genetic Algorithms in Search Optimization and Machine Learning.

Golovkin, I. E., Mancini, R. C., Louis, S. J., Lee, R. W., & Klein, L. (2000, July). Multi-criteria search and optimization: An application to x-ray plasma spectroscopy. In

Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No. 00TH8512) (Vol. 2, pp. 1521-1527). IEEE.

Greenberg, J. A., Dobrowski, S. Z., & Vanderbilt, V. C. (2009). Limitations on maximum tree density using hyperspatial remote sensing and environmental gradient analysis. Remote Sensing of Environment, 113(1), 94-101.

Hamraz, H., Contreras, M. A., & Zhang, J. (2016). A robust approach for tree segmentation in deciduous forests using small-footprint airborne LiDAR data. International Journal of Applied Earth Observation and Geoinformation, 52, 532–541. https://doi.org/10.1016/j.cageo.2017.02.017

Hanan, E. J., Ren, J., Tague, C. L., Kolden, C. A., Abatzoglou, J. T., Bart, R. R., ... & Adam, J. C. (2021). How climate change and fire exclusion drive wildfire regimes at actionable scales. Environmental Research Letters, 16(2), 024051.

Hartsook, T. (2021). Discovetree-An Automated Tool To Generate Stem Maps From Terrestrial Laser Scanner Point Clouds (Doctoral dissertation, University of Nevada, Reno).

Heumann, B. W. (2011). Satellite remote sensing of mangrove forests: Recent advances and future opportunities. Progress in Physical Geography, 35(1), 87-108.

Hijmans R (2022). _raster: Geographic Data Analysis and Modeling_. R package version 3.5-29, https://CRAN.R-project.org/package=raster

Holmgren, J., Persson, Å., & Söderman, U. (2008). Species identification of individual trees by combining high resolution LiDAR data with multi-spectral images. International Journal of Remote Sensing, 29(5), 1537-1552.

Horn, J., Nafpliotis, N., & Goldberg, D. E. (1994, June). A niched Pareto genetic algorithm for multiobjective optimization. In Proceedings of the first IEEE conference on evolutionary computation. IEEE world congress on computational intelligence (pp. 82-87). Ieee.

Jean-Romain Roussel and David Auty (2022). Airborne LiDAR Data Manipulation and Visualization for Forestry Applications. R package version 4.0.0. https://cran.r-project.org/package=lidR

Ji-Hua, M., & Bing-Fang, W. (2008). Study on the crop condition monitoring methods with remote sensing. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 37(B8), 945-950.

Karlson, M., Reese, H., & Ostwald, M. (2014). Tree crown mapping in managed woodlands (parklands) of semi-arid West Africa using WorldView-2 imagery and geographic object based image analysis. Sensors, 14(12), 22643-22669.

Kattenborn, T., Leitloff, J., Schiefer, F., & Hinz, S. (2021). Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. ISPRS journal of photogrammetry and remote sensing, 173, 24-49.

Ke, Y., & Quackenbush, L. J. (2011). A review of methods for automatic individual tree-crown detection and delineation from passive remote sensing. International Journal of Remote Sensing, 32(17), 4725-4747.

Kellner, J. R., Armston, J., Birrer, M., Cushman, K. C., Duncanson, L., Eck, C., ... & Zgraggen, C. (2019). New opportunities for forest remote sensing through ultra-high-density drone lidar. Surveys in Geophysics, 40, 959-977.

Khanal, S., Kc, K., Fulton, J. P., Shearer, S., & Ozkan, E. (2020). Remote sensing in agriculture—accomplishments, limitations, and opportunities. Remote Sensing, 12(22), 3783.

Lantschner, M. V., & Corley, J. C. (2015). Spatial pattern of attacks of the invasive woodwasp Sirex noctilio, at landscape and stand scales. PLoS One, 10(5), e0127099.

Larsen, M. (2007). Single tree species classification with a hypothetical multi-spectral satellite. Remote Sensing of Environment, 110(4), 523-532.

Li, S., Dao, V., Kumar, M., Nguyen, P., & Banerjee, T. (2022). Mapping the wildland-urban interface in California using remote sensing data. Scientific reports, 12(1), 5789.

Li, W., Guo, Q., Jakubowski, M. K., & Kelly, M. (2012). A new method for segmenting individual trees from the lidar point cloud. Photogrammetric Engineering & Remote Sensing, 78(1), 75-84.

Liang, X., Litkey, P., Hyyppa, J., Kaartinen, H., Vastaranta, M., & Holopainen, M. (2011). Automatic stem mapping using single-scan terrestrial laser scanning. IEEE Transactions on Geoscience and Remote Sensing, 50(2), 661-670.

Liashchynskyi, P., & Liashchynskyi, P. (2019). Grid search, random search, genetic algorithm: a big comparison for NAS. arXiv preprint arXiv:1912.06059.

Lin, Z., & Zhang, G. (2020). Genetic algorithm-based parameter optimization for EO-1 Hyperion remote sensing image classification. European Journal of Remote Sensing, 53(1), 124-131.

Lindberg, E., & Holmgren, J. (2017). Individual tree crown methods for 3D data from remote sensing. Current forestry reports, 3, 19-31.

Liu, J., Liang, X., Hyyppä, J., Yu, X., Lehtomäki, M., Pyörälä, J., ... & Chen, R. (2017). Automated matching of multiple terrestrial laser scans for stem mapping without the use of artificial references. International Journal of Applied Earth Observation and Geoinformation, 56, 13-23.

Louis, S. J., & Johnson, J. (1997, October). Solving Similar Problems Using Genetic Algorithms and Case-Based Memory. In ICGA (pp. 283-290).

Luca Scrucca (2017). On some extensions to GA package: hybrid optimisation, parallelisation and islands evolution. The R Journal, 9/1, 187-206. https://doi.org/10.32614/RJ-2017-008

Marchesini, E., & Farinelli, A. (2020, May). Genetic deep reinforcement learning for mapless navigation. In Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (pp. 1919-1921).

Marín-Buzón, C., Pérez-Romero, A., Tucci-Álvarez, F., & Manzano-Agugliaro, F. (2020). Assessing the orange tree crown volumes using google maps as a low-cost photogrammetric alternative. Agronomy, 10(6), 893.

Moody, T. J., Fites-Kaufman, J., & Stephens, S. L. (2006). Fire history and climate influences from forests in the northern Sierra Nevada, USA. Fire Ecology, 2, 115-141.

Moriya, É. A. S., Imai, N. N., Tommaselli, A. M. G., Berveglieri, A., Santos, G. H., Soares, M. A., ... & Reis, T. T. (2021). Detection and mapping of trees infected with citrus gummosis using UAV hyperspectral data. Computers and Electronics in Agriculture, 188, 106298.

Morsdorf, F., Meier, E., Kötz, B., Itten, K. I., Dobbertin, M., & Allgöwer, B. (2004). LIDAR-based geometric reconstruction of boreal type forest stands at single tree level for forest and wildland fire management. Remote sensing of environment, 92(3), 353-362.

Mumtaz, A., Majid, A., & Mumtaz, A. (2008, October). Genetic algorithms and its application to image fusion. In 2008 4th International Conference on Emerging Technologies (pp. 6-10). IEEE.

Næsset, E., & Bjerknes, K. O. (2001). Estimating tree heights and number of stems in young forest stands using airborne laser scanner data. Remote sensing of Environment, 78(3), 328-340.

Näsi, R., Honkavaara, E., Lyytikäinen-Saarenmaa, P., Blomqvist, M., Litkey, P., Hakala, T., ... & Holopainen, M. (2015). Using UAV-based photogrammetry and hyperspectral imaging for mapping bark beetle damage at tree-level. Remote Sensing, 7(11), 15467-15493.

Nwilo, P., Onyegbula, J., Okolie, C., Daramola, O., Abolaji, O., & Arungwa, I. (2022). Quality Assessment of the 20m SPOT DEM in Nigeria. arXiv preprint arXiv:2205.02946.

Ogden, J. (1985). An introduction to plant demography with special reference to New Zealand trees. New Zealand journal of botany, 23(4), 751-772.

Onishi, M., & Ise, T. (2021). Explainable identification and mapping of trees using UAV RGB image and deep learning. Scientific reports, 11(1), 903.

Paterna, S., Santoni, M., & Bruzzone, L. (2020). An automatic planning and scheduling method based on multi-objective Genetic Algorithms for planetary radar sounder observations. In IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium (pp. 5967-5970). IEEE.

PDAL Contributors, 2022. PDAL Point Data Abstraction Library. https://doi.org/10.5281/zenodo.2616780

Popescu, S. C. (2007). Estimating biomass of individual pine trees using airborne lidar. Biomass and Bioenergy, 31(9), 646-655.

QGIS.org, 2019. QGIS Geographic Information System. QGIS Association. http://www.qgis.org

Qian, C., Liu, H., Tang, J., Chen, Y., Kaartinen, H., Kukko, A., ... & Hyyppä, J. (2016). An integrated GNSS/INS/LiDAR-SLAM positioning method for highly accurate forest stem mapping. Remote Sensing, 9(1), 3.

Queinnec, M., Coops, N. C., White, J. C., Griess, V. C., Schwartz, N. B., & McCartney, G. (2023). Mapping Dominant Boreal Tree Species Groups by Combining Area-Based and Individual Tree Crown LiDAR Metrics with Sentinel-2 Data. Canadian Journal of Remote Sensing, 49(1), 2130742.

R Core Team (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

RapidLasso GmbH. (2019). Lastools version 191111.

Renslow, M., Greenfield, P., & Guay, T. (2000). Evaluation of multi-return LIDAR for forestry applications. US Department of Agriculture Forest Service-Engineering, Remote Sensing Applications. http://www. ndep. gov/USDAFS/LIDAR. pdf [Consulta: 12 de marzo de 2009].

Riegl Laser Measurement Systems GmbH. (2020). Riscan pro 2.0.

Roussel J (2022). _lidRplugins: Extra functions and algorithms for lidR package_. R package version 0.4.0.

Roussel, J.R., Auty, D., Coops, N. C., Tompalski, P., Goodbody, T. R. H., Sánchez Meador, A., Bourdon, J.F., De Boissieu, F., Achim, A. (2020). lidR : An R package for analysis of Airborne Laser Scanning (ALS) data. Remote Sensing of Environment, 251 (August), 112061. <doi:10.1016/j.rse.2020.112061>.

Ryu, Y., Lee, G., Jeon, S., Song, Y., & Kimm, H. (2014). Monitoring multi-layer canopy spring phenology of temperate deciduous and evergreen forests using low-cost spectral sensors. Remote Sensing of Environment, 149, 227-238.

Sanger, M. C., & Barnett, K. (2021). Remote sensing and indigenous communities: Challenges and opportunities. Advances in Archaeological Practice, 9(3), 194-201.

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 Journal of Photogrammetry and Remote Sensing, 170, 205-215.

Scrucca, L. (2013). GA: a package for genetic algorithms in R. Journal of Statistical Software, 53(1), 1-37. https://doi.org/10.18637/jss.v053.i04

Shafri, Helmi ZM, and Nasrulhapiza Hamdan. "Hyperspectral imagery for mapping disease infection in oil palm plantationusing vegetation indices and red edge techniques." American Journal of Applied Sciences 6, no. 6 (2009): 1031.

Silva, C. A., Hudak, A. T., Vierling, L. A., Loudermilk, E. L., O'Brien, J. J., Hiers, J. K., Khosravipour, A. (2016). Imputation of Individual Longleaf Pine (Pinus palustris Mill.) Tree Attributes from Field and LiDAR Data. Canadian Journal of Remote Sensing, 42(5), 554–573. https://doi.org/10.1080/07038992.2016.1196582.

Slonecker, E. T., Shaw, D. M., & Lillesand, T. M. (1998). Emerging legal and ethical issues in advanced remote sensing technology. Photogrammetric engineering and remote sensing, 64(6), 589-595.

Stanton, A. E., & Dailey, S. N. (2007). Pre-treatment and partial-treatment forest structure and fuel loads in the Lake Tahoe Basin Management Unit. BMP Ecosciences, San Francisco, California.

Tang, J., Chen, Y., Kukko, A., Kaartinen, H., Jaakkola, A., Khoramshahi, E., ... & Hyyppä, H. (2015). SLAM-aided stem mapping for forest inventory with small-footprint mobile LiDAR. Forests, 6(12), 4588-4606.

Tao, S., Labrière, N., Calders, K., Fischer, F. J., Rau, E. P., Plaisance, L., & Chave, J. (2021). Mapping tropical forest trees across large areas with lightweight cost-effective terrestrial laser scanning. Annals of Forest Science, 78(4), 103.

Tonolli, S., Dalponte, M., Vescovo, L., Rodeghiero, M., Bruzzone, L., & Gianelle, D. (2011). Mapping and modeling forest tree volume using forest inventory and airborne laser scanning. european Journal of forest Research, 130, 569-577.

Ulku, I., Akagündüz, E., & Ghamisi, P. (2022). Deep semantic segmentation of trees using multispectral images. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 15, 7589-7604.

United States, U.S. Department of Agriculture. Lake Tahoe Basin Mgt Unit - Home. Forest Service, 2009.

United States, U.S. Department of Agriculture. Plumas National Forest – About the Forest. Forest Service, 2012.

Vega, C., Hamrouni, a., El Mokhtari, S., Morel, J., Bock, J., Renaud, J.-P., … Durrieu, S. (2014). PTrees: A point-based approach to forest tree extraction from lidar data. International Journal of Applied Earth Observation and Geoinformation, 33, 98–108. https://doi.org/10.1016/j.jag.2014.05.001

Wade, L. E. (2022). Deriving Novel Allometric Equations for Northern Sierra Nevada Trees Using Terrestrial Laser Scanning (Doctoral dissertation, University of Nevada, Reno).

Weiss, U., Biber, P., Laible, S., Bohlmann, K., & Zell, A. (2010, December). Plant species classification using a 3D LIDAR sensor and machine learning. In 2010 Ninth International Conference on Machine Learning and Applications (pp. 339-345). IEEE.

Wulder, M., Niemann, K. O., & Goodenough, D. G. (2000). Local maximum filtering for the extraction of tree locations and basal area from high spatial resolution imagery. Remote Sensing of environment, 73(1), 103-114.

Yang, M., Mou, Y., Liu, S., Meng, Y., Liu, Z., Li, P., ... & Peng, C. (2022). Detecting and mapping tree crowns based on convolutional neural network and Google Earth images. International Journal of Applied Earth Observation and Geoinformation, 108, 102764.

Yu, Q., Gong, P., Clinton, N., Biging, G., Kelly, M., & Schirokauer, D. (2006). Object-based detailed vegetation classification with airborne high spatial resolution remote sensing imagery. Photogrammetric engineering and remote sensing, 72(7), 799.

Zhang, W., Wan, P., Wang, T., Cai, S., Chen, Y., Jin, X., & Yan, G. (2019). A novel approach for the detection of standing tree stems from plot-level terrestrial laser scanning data. Remote sensing, 11(2), 211.

Zhao, K., Suarez, J. C., Garcia, M., Hu, T., Wang, C., & Londo, A. (2018). Utility of multitemporal lidar for forest and carbon monitoring: Tree growth, biomass dynamics, and carbon flux. Remote Sensing of Environment, 204, 883-897.

Zong, X., Wang, T., Skidmore, A. K., & Heurich, M. (2021). The impact of voxel size, forest type, and understory cover on visibility estimation in forests using terrestrial laser scanning. GIScience & remote sensing, 58(3), 323-339.

Conclusion

This thesis demonstrated the use of genetic algorithms (GAs) for optimizing the performance of six (6) individual tree crown recognition (ITC) models for mapping trees using airborne LiDAR (Light Detection and Ranging) at a landscape level. The GA-tuned parameters outperformed the default values for 5 out of 6 of them, evaluated based on their F-scores. Also, we found out that each ITC model was generally poor at predicting the actual number of trees on site, with either gross overestimation or underestimation errors. However, they had better performances in detecting large trees than a mix of trees of different sizes. The GA-optimization of these ITCs was limited by the inherent capabilities of the original models themselves, and accurate tree crown mapping from airborne LiDAR probably still remains an unsolved remote sensing problem.