Using Drones and Remote Sensing to Understand Forests with Limited Groundtruth Data

David Russell

CMU-RI-TR-23-34

August 15, 2023

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee:
Professor David Wettergreen, *chair*
Professor George Kantor
Professor Marija Popović, *University of Bonn*
Kshitij Goel

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Robotics.

To my friends, family, Athena, and Marvel

Abstract

Drones and remote sensing can provide observations of forests at scale, but this raw data needs to be interpreted to further our scientific understanding and inform effective management decisions. This thesis studies two problems under the realistic constraint of limited domain-specific training data: tree detection for understanding carbon sequestration and vegetation mapping for forest fire mitigation.

For tree detection, we process the drone data using structure from motion and then register it to remote sensing imagery. Then, we compare different strategies for using a deep learning detector with these modalities and limited training data. For vegetation mapping, we show that we can localize fuel that causes forest fires using image-based semantic segmentation trained on very few examples and LiDAR-based geometric reasoning. Finally, we introduce RAPTORS, a novel algorithm that plans where to collect sparse drone observations based on existing remote sensing data. We show that training a remote sensing-based vegetation classification model on observations from RAPTORS is more effective at identifying rare classes than training on observations from a coverage-based approach. Overall, these experiments show how using machine learning, data harmonization across scales, and intelligent sampling can facilitate automated forest understanding with limited training data.

Acknowledgments

I am sincerely grateful to Prof. George Kantor, Prof. Marija Popovic, and Kshitij Goel for taking the time to serve on my committee for their time and insightful feedback. And especially to my advisor, Prof. David Wettergreen, for all the suggestions, direction, encouragement and understanding. I appreciate both your willingness to let me explore and ability to direct my focus to what matters.

To my labmates, Srini Vijayarangan, Rohan Zeng, Ananya Rao, Maggie Hansen, and Abby Brietfield. Thank you for all the discussions about research and real life. You taught me a lot and made this far more fun.

Much of this work happened in close collaboration with the SafeForest team: Dr. Francisco Yandun, Winnie Kuang, and Duda Andrada. I appreciate the numerous times you have provided me useful data, helped me debug code, or clarify my thoughts.

To Dr. Derek Young and Dr. Mike Koontz at Open Forest Observatory, I appreciate your perspectives on how this work really gets done.

To the directors of the Robotics Institute Summer Scholars program, Prof. John Dolan and Rachel Burcin. Thank you for going above and beyond to make opportunities for me and so many others. I would not have been here without you.

Funding

This material is based upon work supported by the following sources:

- The Safeforest Project – Semi-Autonomous Robotic System for Forest Cleaning and Fire Prevention (CENTRO-01-0247-FEDER-045931), co-financed by FEDER, COMPETE 2020, by the CENTRO2020 Program and by Fundação para a Ciência e Tecnologia (FCT) under the CMU Portugal Program.
- The AI Institute for Resilient Agriculture (AIIRA), supported by the AI Research Institutes program supported by NSF and USDA-NIFA under AI Institute: for Resilient Agriculture, Award No. 2021-67021-35329
- The National Science Foundation (NSF) STTR Phase I: Registration of Below-Canopy, Above-Canopy, and Satellite Sensor Streams for Forest Inventories Award No. 2234077

Views expressed are those of the author and do not necessarily represent those of the funding agency.

Contents

When this dissertation is viewed as a PDF, the page header is a link to this Table of Contents.

List of Figures

List of Tables

Chapter 1

Introduction

Forests impact many aspects of our life on Earth, such as removing CO_2 from the atmosphere, purifying water, moderating local temperature, and supporting human livelihoods. Unfortunately, forests are under threat from a variety of sources including climate change, invasive species, fire, and direct human pressures [50]. This is causing forests to change at an unprecedented rate. In light of these rapid changes, it is critical that we have up-to-date information to inform decisions such as habitat preservation, sustainable timber operations, forest fire mitigation, and carbon sequestration. In this thesis, we develop approaches motivated by forest fire mitigation and carbon sequestration, but these methods aim to be generic enough to also be suitable for other applications.

There are many sources of data that can be used to inform forest management. This thesis studies three representative sources of data: manual field measurements, data from drones, and remote sensing imagery. The manual measurements are accurate and granular but fail to provide information at scale. Conversely, remote sensing data can provide information at scale, but lacks fine-grained spatial detail and is challenging to interpret. Drone data strikes a middle ground. The goal of this work is to develop techniques that leverage all three sources of data to produce insights about forests that are both accurate and scalable.

1.1 Research Questions

We propose the following concept for an intelligent future forest management system: First, a forester goes to a new area and collects a limited number of representative field observations for a task of interest, such as vegetation mapping or detecting trees, and then surveys the same areas with a drone. Second, the system uses these observations to train a machine learning model to perform this task on new drone data. Third, the system uses remote sensing data to propose a set of representative locations to visit with the drone. The forester conducts the drone flights and ingests this data into the system. Fourth, predictions are generated for this new data using the previously trained model. Finally, these predictions are used to train a model that generates predictions from remote sensing data for the entire region. This concept leverages the forester's domain knowledge to obtain ground truth measurements and uses both drones and remote sensing to incrementally scale these observations to a large region.

As initial steps toward this concept, we propose the following three research questions:

- How can data from field surveys, drones, and remote sensing be integrated to accurately detect trees at scale?
- How can drones be used to classify vegetation types in a large forested region?
- How can sparse drone surveys be planned to provide the most diverse and informative measurements about a region?

1.2 Methods

A challenge in conducting this research is the limited availability of applicable datasets. Therefore, we begin this work by collecting data in a diverse set of forests using both a commodity drone and a custom multi-sensor payload. We extract geometric information from this data using two different approaches, structure from motion and simultaneous localization and mapping.

To study tree detection, we generate an orthomosaic from the drone data using structure from motion and then label the location of a small number of trees. Then,

we register this data with aerial remote sensing data. Finally, we explore several strategies for applying a deep learning tree detector with different combinations of these modalities.

For vegetation mapping, we use a LiDAR-based simultaneous localization and mapping approach to understand the structure of the scene. Then we determine the types of vegetation using an image-based semantic segmentation approach. These predictions are aggregated into a 3D representation that captures both the structure of the environment and what type of vegetation is present at each location.

To plan informative drone flights, we begin with remote sensing data for the region. Using an unsupervised method, we extract informative texture features. These features are then used to model the uncertainty of unobserved points given a set of observations. Using this uncertainty model, we incrementally plan the set of locations to observe over an entire drone flight. This plan seeks to minimize the uncertainty of the entire region while still maintaining feasibility under the battery constraints.

1.3 Contributions

We find that structure from motion is a powerful tool for processing over-canopy drone flights using only GPS-tagged images and requires minimal hand-tuning. In an under-canopy setting, a LiDAR-based simultaneous localization and mapping approach demonstrated better performance but this requires more complex sensors and site-specific parameter tuning.

Our experiments on tree detection are consistent with previous results showing that trees can be easily detected in drone data and that site-specific fine-tuning with a small amount of data only yields a small improvement. We find that applying the model that was trained on a diverse set of drone data to aerial remote sensing data yields poor performance. Fine-tuning the model for remote sensing data results in a moderate increase in quality but the quality of these detections is still insufficient for most tasks. Training a remote sensing model using predictions from the drone yields an improvement over the pre-trained model. However, training on a small amount of labeled data is still more effective.

In the fuel mapping setting, we find that modern transformer-based semantic

segmentation networks can generate adequate predictions by training on only a small number of labeled images from the same scene. We show that this network can be used in combination with the LiDAR-based SLAM to build a metric-semantic map of the world that captures both the geometry and the classification of each region. This allows us to conduct vegetation class mapping automatically with a drone.

Finally, we show that random kernels and dimensionality reduction yields informative unsupervised features. These allow us to perform land-cover mapping with a simple k-nearest neighbor classifier. We evaluate the informative path planning approach on a variety of locations and show that even though the paths look qualitatively reasonable, the quality of the improvement is small compared to the variability between the different trials. This highlights the challenge of building generalizable informative path planning approaches and suggests that further characterization of what samples are informative is required.

We conclude with suggestions for how to unify multiple themes from these experiments. We recommend further studying the effects of spatial resolution on tree detection by simulating a range of resolutions from high-resolution drone data. We suggest that semantic mapping can be made more robust and accessible by using structure from motion rather than simultaneous localization and mapping to predict geometry. We propose that the work on semantic mapping, informative path planning, and remote sensing predictions can be integrated into a comprehensive field experiment to directly address the feasibility of the conceptualized forest understanding system. Finally, we highlight the need for more interdisciplinary datasets and open-source software to foster further work in this space.

Chapter 2

Background

2.1 Application Areas

We focus our efforts on two application areas: predicting carbon sequestration in forests and mapping vegetation types for forest fire mitigation. These provide representative examples to motivate vegetation type mapping and tree detection, respectively.

2.1.1 Forest Carbon Sequestration Prediction

The first problem we explore is assessing carbon sequestration in forests. Forests are a massive sink of carbon, so protecting and managing our forests is a critical tool in the fight against climate change [41]. Clearing forests can result in substantial CO_2 emissions so it is especially important to keep existing forests intact. One tool to incentivize this is *carbon credits*, which are payments to the landowner to keep carbon sequestered. These payments are often made by businesses or governments who have pledged to meet certain emission targets but cannot reduce their CO_2 output to fulfill them immediately. Instead, they *offset* these emissions by paying to have a commensurate amount of CO_2 emissions prevented. A key rationale for carbon credits is that some industries and activities are more challenging to de-carbonize than others, so it makes sense to direct funding toward the easiest solutions to achieve the largest near-term emission reductions. It is expected that demand for carbon

credits will rise by a factor of 50 by 2050 according to Blaufelder et al. [13].

Unfortunately, there are many factors that make it challenging to guarantee that a forest-based carbon credit will have the desired impact and actually result in the claimed reduction in CO_2 emission. Some considerations are whether issuing the credit actually results in behavior change [39] or that carbon will later be released by uncontrollable factors such as wildfire [54]. Another concern is that the established auditing approaches may overestimate the amount of carbon stored in a plot of land, which means that even in the best-case scenario, the desired amount of carbon sequestration is not being obtained. There are a variety of works showing that systematic over-crediting is common [9, 107] due to biased modeling efforts or bias in human estimations. Technology can play a role in this area by improving upon simple regional models with empirical and site-specific information.

A common method for accurately estimating the carbon content of a forest is by taking a tree-centric approach. Each tree is located and the carbon content of each one is estimated individually. The per-tree carbon content can be regressed from phenotypic values such a basal (crown) area [100] or predicted directly from images using machine learning [80]. The carbon content of the landscape can then be estimated by summing up the per-tree contributions. Our work on this topic focuses on the first stage of the carbon estimation process: detecting trees.

2.1.2 Forest Fire Mitigation

Destructive forest fires have increased dramatically over the past several decades [17, 62, 89]. This is due in large part to climate change, which leads to hotter and drier weather along with stronger winds [62]. Climate change has also led to increased forest mortality from pests expanding their range, such as the mountain pine beetle in the Western US [52]. Humans have contributed directly to fires by suppressing small fires which causes a build-up of flammable vegetation over time. Finally, there is an increase in ignition sources from careless human activity and infrastructure such as power lines. At the same time that fires are growing more common and destructive, more people live in close proximity to forests, furthering the risk of property loss, injury, and death. The ecological consequences of fire are also increasingly dire. Historical fires were a natural part of some ecosystems and vegetation was able to

regenerate due to surviving trees and un-burned seeds. The intensity of modern fires destroys all vegetation and seeds, making it harder for regions to regrow. This can lead to erosion and eventual transition from forest to grassland.

It is becoming increasingly clear that reactive firefighting is insufficient to combat modern forest fires and preemptive mitigation efforts are also required [62]. One way to actively reduce the risk of destructive fires is by removing dense understory vegetation in a process termed fuel management [3, 37, 108]. This is a challenging problem due to the sheer area of forested land and the limited resources currently put toward preemptive efforts [62].

Fuel management is physically demanding and requires specialized knowledge, which has led to increasing concerns about labor shortages [21]. A recent robotics project proposed a multi-robot team that could autonomously remove vegetation [25]. In their work, an unmanned ground vehicle can traverse the environment and mechanically grind vegetation, rendering it a less potent fuel source. This robot only has an understanding of its local environment, so the proposed concept relies on drones to map the environment beforehand. These drones determine both the geometric structure of the scene and the location of the fuel. This helps the robot avoid obstacles and determine where to go, respectively. Our work tackles this fuel mapping problem from the drone and proposes to extend this work to broader regions by predicting vegetation locations from remote sensing data.

2.2 Sources of Data for Forestry

Most of our current forest understanding is at a broad scale [71], but making intelligent forestry decisions requires granular information about the current state of the forest [46, 99]. The previous section highlights two examples where this is critical. Unfortunately, it is challenging to obtain information that is both accurate and covers a large enough region to inform management decisions at scale. We look at three sources of data that can inform forest management: observations from manual field work, images from small uncrewed aerial vehicles (UAVs or "drones"), and optical remote sensing data taken from aircraft or satellites. These three sources of data represent a trade-space between high interpretability from manual observations and high scalability from remote sensing.

2.2.1 Manual Plot Measurements

Understanding forests is still a largely manual process where foresters measure various quantities such as tree height, diameter, density, and species while in the field. In commercial contexts, this process is called a timber cruise [92] and in ecology, it is called a forest inventory [104]. Since this process is laborious, the established practice is to only take measurements at points or plots distributed throughout the environment [101]. Choosing where to sample these plots is an important component of obtaining accurate and unbiased estimates for a region. Determining how to balance the size versus the number of plots and designing a sampling procedure both require substantial domain expertise.

2.2.2 Drone Surveys

Drones equipped with cameras are increasingly used in forestry [27, 98]. This is driven by their comparatively low cost, ease of use, and flexibility. As described by Tang et al. [98], drones fill a crucial operational gap by capturing higher-resolution data than crewed aircraft or satellites. Therefore, they are helpful for a wide variety of forestry tasks that require granular information.

Cameras are incorporated in nearly every drone because of their usefulness, low cost, and weight. Drones are often equipped with a commodity-grade GPS for navigation, which means that the captured images can be tagged with an approximate absolute location. Even though drones can be flown manually, it is common in forestry applications to use software such as QGroundControl [77] or DroneDeploy [32] to plan a flight pattern. The user defines the perimeter of the area and sets parameters such as the survey pattern, altitude, and image density. The area that a drone can cover is often several acres but varies based on the type of drone and the type of flight plan. Similarly, the spatial resolution of the data varies but is on the order of centimeters per pixel. While drones are a powerful tool, the raw data from them must generally be processed in a domain-specific manner to obtain the ecological information that is required by the end user.

2.2.3 Remote Sensing

Remote sensing data is captured by sensors onboard satellites or airplanes and is rapidly becoming a critical tool for environmental monitoring [73]. This is largely because these sensors observe large—potentially global—regions and much of this data is available freely to the public. Remote sensing data can be from many modalities, such as light detection and ranging (LiDAR) [11], synthetic aperture radar (SAR) [43], and electro-optical (EO) data. In this work, we focus on electro-optical data, since it is prevalent and easy to interpret. This data is conceptually very similar to images taken by a traditional camera. While most consumer cameras only capture red-green-blue (RGB) information, remote sensing instruments often capture more distinct spectral bands, termed multi- or hyper-spectral data depending on the number. Another consideration is that remote sensing data often undergo many post-processing steps before being released. These include removing atmospheric effects, stitching images together, and re-sampling the data to lie on an axis-aligned grid. Optical remote sensing data has been used for numerous forestry applications such as mapping forest coverage [44] and mapping forest type [56]. Unfortunately, this technology has its limits, such as low spatial resolution which precludes granular analysis. For example, the forest extent mapping conducted by Hansen et al. [44] was only available at 30-meter resolution, because that was the resolution of the input data from the LandSat [68] satellite. This low resolution also means that multiple classes may be contained in one pixel, which further complicates automated analysis.

2.3 Interpreting Forestry Data

The previous section highlights three types of data that can be obtained from forests. Manual forest inventories directly collect interpretable data, such as tree species and size, that can be statistically extrapolated to a larger region. The data from drones and remote sensing must be interpreted before it can be used to inform management decisions. As described in Section 2.1.2, some applications require a detailed geometric understanding of the environment while others need a map of what type of vegetation is where.

2.3.1 Understanding the Geometry of Scenes

There are two approaches to understanding the geometry of forests that are common in a robotics context. The first approach, structure from motion, only requires a set of images about the scene but processing can only take place after the mission is complete. The second approach, simultaneous localization and mapping (SLAM), can be run online while the drone is flying, but often requires multiple complementary sensors for good results.

Structure from Motion

In general, commodity drones produce only monocular images with potentially a low-accuracy GPS and orientation estimate. A common approach for estimating geometry from this type of data is photogrammetry, also known as structure from motion or 3D reconstruction. Preliminary reconstructions of realistic large-scale scenes began with academic work such as [1]. Over the last decade, numerous commercial and open sources software have been developed for this task, such as Agisoft Metashape [2] and COLMAP [90, 91], respectively.

The implementation details vary by application and assumption, but a common pipeline is the following: first, distinctive features are detected in each image. These represent small patches of pixels that are likely to be informative, such as corners and edges. Then, features are matched between images based on the local appearance. Given these corresponding points between images, multiple quantities can be estimated. The first is the location of these matched image points in the 3D space using triangulation between the cameras. The second is the location of the cameras. Finally, if the camera isn't accurately calibrated, it is common to estimate the *intrinsic parameters* which describe how points in the world are projected onto the image. Given the interplay between all of these elements, it's critical to estimate them together in a global joint optimization. A widely-used class of techniques for solving this problem are termed bundle adjustment [102]. After the locations of the cameras have been estimated, it is possible to construct a dense point cloud or mesh representation of the scene, an example of which can be seen in Figure 2.1.

Photogrammetry has been used in a variety of recent works on understanding forests [26, 97]. A notable work in this space is that of Young et. al. [114]. This

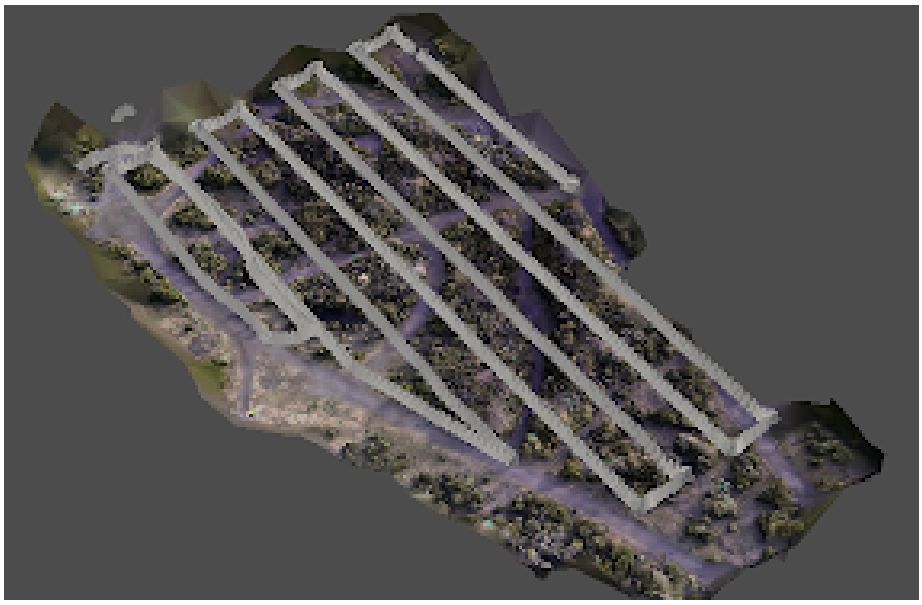


Figure 2.1: An example 3D reconstruction from Agisoft Metashape [2] with the camera locations from the drone survey visualized.

work studies the impacts of different drone fight patterns and Metashape processing parameters on the quality of tree detection in complex coniferous forests. They analyze thousands of different configurations to propose a flight pattern and set of photogrammetry parameters that can be used in other tree detection applications.

Simultaneous Localization and Mapping

In settings where a drone is operating autonomously in complex environments such as under the canopy, it is important that it understands where it is in relation to obstacles in real time. This is necessary for the robot to can plan a trajectory and avoid collisions. The problem is known in robotics literature as simultaneous localization and mapping (SLAM) [34] since it involves solving two challenging problems at once. The first is localization, where the robot must determine its position within a known map. This is especially important in a forestry setting because GPS can be unreliable under forest canopies. The second is mapping, which involves building a 3D representation of the world using sensor data and the current location of the robot. In a new environment, the robot does not have a prior map, so these two interconnected tasks must be completed simultaneously.

In general, SLAM relies heavily on optimization approaches to jointly estimate the structure of the map and the location of the robot over time. Many modern approaches use factor graphs [31], which are an efficient formulation that allows previous estimates to be continuously refined as new information is obtained. A wide variety of SLAM systems have been proposed for different sensors and environmental settings. Many approaches use a LiDAR and an inertial measurement unit (IMU) because the former provides explicit 3D information and the latter provides an accurate estimate of the motion over a short time horizon even in uninformative environments. A commonly-used approach is LIO-SAM [93]. This approach estimates the motion of the system by registering consecutive LiDAR scans, after using the IMU to remove distortion and provide an initial estimate to initialize the matching process. Another approach is LOAM [115], which extracts geometric features such as corners and edges from the point LiDAR scans to provide more informative correspondences. This approach is more suitable for built environments than forests since these geometric features are less common in unstructured natural environments. However, an extension to LOAM, termed SLOAM [18], is designed specifically for forests. In SLOAM, they detect tree trunks from the LiDAR scans and use these trunks as landmarks to improve the localization. This approach shows strong results in forests but relies heavily on having a high-quality tree detection algorithm.

2.3.2 Understanding the Content of Images

Drones and remote sensing can collect a vast amount of data about the environment. Before this data is directly useful to land managers, it is important to extract quantities such as the location and size of trees or the types of vegetation in each region. Automated processing methods can free domain experts from the laborious task of interpreting data by hand. There has been a steady shift from methods that are hand-designed to those which learn from data. Supervised machine learning is a class of methods where the model is provided both the raw data and the correct interpretation. The model that is developed from these training examples can be used to generate predictions on new data. Over the last decade there has been an explosion of approaches relying on deep learning [65], which is a subset of machine learning using models with a large number of parameters that have multiple hierarchical processing

steps. The parameters of these models are updated or *trained* by an iterative process that seeks to minimize the error or *loss* between the predictions and corrected labels. Because of the high number of parameters, these models often require large amounts of training data to generalize well to new data.

Object Detection

Object detection is the problem of identifying the location and shape of objects within an image. The shape of the object is often represented by an axis-aligned rectangle or a pixel-wise mask. These approaches can be trained to solely identify one type of object or identify and distinguish objects of multiple classes. In a forestry context, this could refer to both predicting the location and species of a tree.

This problem has been extensively studied by the computer vision community over the last two decades. Early work was conducted by Dalal and Triggs [28], where they proposed a robust solution to pedestrian detection. Their approach leveraged well-engineered feature extraction using local changes in intensity and a support vector machine (SVM) classifier. Many approaches built on this concept, often still focusing heavily on extracting informative features from images. A seminal paper in 2012 by Krizhevsky et. al. [61] showed that learning informative features from data using convolutional neural networks (CNNs) provided superior results to hand-crafted features on an image classification task. This sparked a trend of CNN-based object detection approaches. An early approach that is still commonly used today is Faster R-CNN [81]. This relies on a multi-stage approach, where the first stage predicts many rectangular candidate object locations. The second stage determines whether the proposal is indeed an object, predicts a refined bounding box, and optionally classifies the type of object. This approach was built upon by Mask R-CNN [45], which predicts a mask representing which pixels are part of the object instead of a rectangle. In contrast to the two-stage approaches, RetinaNet [66] is an efficient single-stage approach that is especially suitable for dense objects, such as trees in a forest. In this work, the authors develop FocalLoss, a penalization strategy that focuses primarily on hard examples during network training.

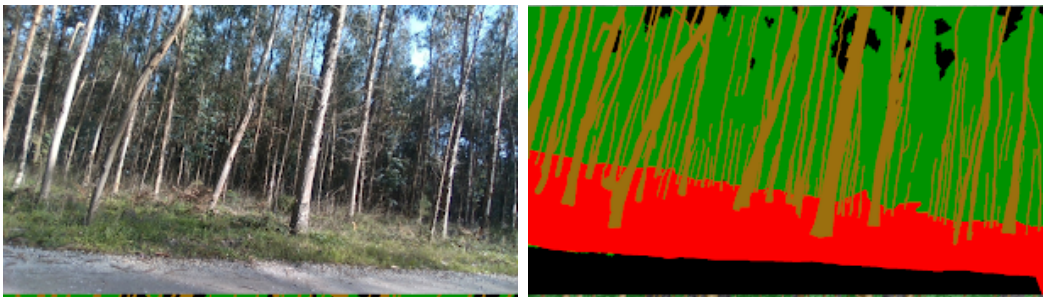


Figure 2.2: A visualization of the goal of semantic segmentation. The input image is on the left, and the desired output is on the right, color-coded by class. Red is understory fuel, green is canopy, brown is trunk, and black is background such as bare earth and sky.

Semantic Segmentation

Semantic segmentation is the task of assigning a classification label to every pixel in an image. In the forestry domain, these classes could be broad, such as trees, shrubs, and grasses, or more granular, such as different species of trees. An example image of this task can be seen in Figure 2.2.

An early work on semantic segmentation with deep learning was Fully Convolutional Network [95], which took early insights from image classification and adapted them to give per-pixel class predictions. Shortly following this was U-Net [85], which had an encoder-decoder architecture with skip connections to preserve high-resolution details. A wide variety of approaches have been developed since then, with slight variations on these initial concepts. There has been a recent shift toward using transformers [105] which has resulted in work such as SegFormer [110] and SegNext [42]. SegFormer is an especially compelling work because the authors conducted evaluations showing that the model generalizes well to data that looks different than what it was trained on. This is useful in the forestry context where there may be limited labeled data to train on and it is not fully representative of the entire scene. The goal behind SegNext is to provide the same level of performance as approaches such as SegFormer, but do so with less technical complexity and faster run-time. This is especially useful for performing semantic segmentation with limited computational resources, such as autonomous systems or laptops in the field.

2.4 Summary

This section summarizes the application areas for our work: forest fire mitigation, and carbon sequestration estimation. Both of these domains require accurate, granular, and scalable information about the state of the forest. We summarize the tradeoffs between data from manual field work, drone surveys, and remote sensing. Accurate—but small scale—information is provided directly by manual surveys, but drone and remote sensing data must be automatically processed to provide insights at scale. Two methods are common for understanding the geometric structure of the world from drone data: one that requires only simple sensors but requires offline processing and another that uses more complex sensors but can generate a map in real time. Deep learning has become a common tool for interpreting the content of scenes and is applicable for both vegetation segmentation tasks and individual tree detection. In this thesis, we use geometric reasoning to understand the structure of these scenes. We then train deep learning models for drone data using small amounts of ground truth information taken from the region and apply these models to the whole scene.

Chapter 3

Related Work

3.1 Detecting Trees from Data at Multiple Scales

A key first step in many ecological modeling applications is understanding the location and extent of individual trees. Because of this, the problem of detecting trees in drone and satellite data has received significant attention. There are numerous approaches for this task and they can be categorized based on whether they use geometric or visual information about the scene.

Geometric approaches take different data as input. A common input is a canopy height map (CHM), which is a 2D top-down representation where each location has a height. This representation is used by Popescu et al. [74] and they apply a sliding window filter to identify tree locations. Other works use point clouds derived from LiDAR, which provides full 3D information about the scene. One such approach by Xiao et al [109] uses the mean shift algorithm to identify clusters of points corresponding to a given tree. This work provides a thorough assessment of the design considerations of using this algorithm.

There are also a diverse set of approaches for detecting trees in visual data. For satellite data, they often rely on ad hoc methods or generic object detection tools provided by proprietary software. For example, Hulet et al. [49] used a multi-resolution segmentation algorithm [8] to segment Pinon and Juniper trees from one-meter resolution aerial data from the National Aerial Imagery Program (NAIP) [103]. This was followed by morphological operations that were manually tuned to be

site-specific. For drone data, there has been an increasing trend toward using deep learning. One widely-used approach is DeepForest [106], a tree detection approach based on RetinaNet [66]. This model was trained on a diverse set of annotations from the National Ecological Observation Network sites [55] across the US. This model has been shown to generalize well to a variety of settings and can be further improved by fine-tuning on a small number of annotations from the local region. Importantly, this model is released in a sophisticated Python package that handles common tasks related to pre-processing, training, inference, and visualizations. One limitation is it only predicts axis-aligned bounding boxes, rather than more detailed representations such as masks. A more recent approach called DetectTree2 [10] addresses this limitation by training a Mask-RCNN [45] model to predict tree boundaries. The limitation of this approach is it was trained on much less data due to the scarcity of mask annotations and has less developed software infrastructure.

The goal of our work is not to directly improve upon these techniques but rather to characterize the performance of existing methods. Specifically, we are interested in applying the same deep learning model to both drone and remote sensing data. This will allow us to quantify the difference in quality between these two sources of data. Similar themes are addressed by Fraser and Gongalton [38], where they compare the quality of species classification and tree detection from drone and NAIP data. For tree detection, they use different learning-free algorithms for each modality. Similar to Hulet et al., they use a multi-resolution segmentation algorithm [8] for detecting trees in NAIP data. For the drone data, they use a marker-controlled water-shed segmentation technique [19]. In our work, we explore deep learning, rather than classical approaches, because it generally achieves high performance and can be easily adapted to a given domain by fine-tuning. We also aim to use the same approach on both modalities and control for as many confounding factors as possible, rather than using different methods.

3.2 Mapping Forest Fire Fuel

The goal of this work is to localize the vegetation that could become fuel for a wildfire using a drone. This is motivated by the concept for a forest management system proposed by Couceiro et al. [24], in which a team of drones provide situational

awareness to an automated ground vehicle. This vehicle is equipped with a mechanical attachment that can grind vegetation and render it much less flammable. The drones are much more maneuverable than the ground vehicle, and can thereby quickly inform it where fuel is and what regions are traversable. As a first step toward this ambitious system, the same team implemented a fuel mapping approach onboard an automated Bobcat, a small, skid-steered utility vehicle commonly used in forestry [6]. This approach used a LiDAR to obtain 3D information about the world and a multi-spectral camera to interpret what is fuel. Using deep learning, they identified fuel clump locations and then localized them in 3D using interpolated LiDAR information. A shortcoming of the proposed approach is that it represents the clump of fuel as a single point. In our work, we wish to represent the full 3D structure of the fuel, as well as other classes of objects in the environment, to provide more context to the ground vehicle.

This type of problem is often termed *semantic mapping* in the robotics community and Kostavalis et al. [59] provides a review of methods for this task. The goal of these approaches is to build a model of the world that captures both the geometry of the scene and the classification of each part of the scene. To complete this task, most approaches rely on an external localization or SLAM to estimate the position and orientation of the robot.

One common framework for semantic mapping is Kimera [86], which relies on data from stereo cameras as input. Using estimated depth from the stereo camera, this method builds a mesh-based representation of the environment. Semantic classification information is added to the mesh and both the geometry and classification can be updated as new information arrives. We hypothesized that Kimera would be poorly suited to forest environments because it was primarily tested on built scenes with large, solid objects. In contrast, forested environments have many highly-textured surfaces and regions, such as canopies, that are not entirely solid. Therefore, we expected that a mesh-based representation would struggle to capture these extremely fine details.

An alternate class of approaches uses sensors that capture explicit 3D information, such as LiDAR or RGB-D cameras. A modular approach for RGB-D semantic mapping is presented by Zhang and Fillat [113]. In this work, they use an image-based semantic segmentation approach to identify the different classes in the scene.

Then, they identify the location of each pixel in 3D space relative to the camera using the depth channel from the RGB-D sensor. Each point is assigned a class from the semantic segmentation image. This local point cloud is transformed into the global coordinate system using the estimated location and orientation of the sensor from the SLAM system. Finally, the information from each semantic point cloud is added to an octomap [48] representation, which is an efficient probabilistic volumetric representation. This octomap is updated as new information is observed. Our previously-published works [7, 88] extend this method to use a LiDAR instead of RGB-D and applies it to the fuel mapping task from drone data.

3.3 Planning Informative Drone Surveys

In many forestry applications, the region of interest is commonly substantially larger than what is feasible to survey, either by hand or even with a drone. In practice, foresters select a small set of plots to visit and extrapolate from these sparse observations to the entire region. These plots are chosen using expert knowledge of the region to be diverse and representative. Despite the ability of drones to cover much larger regions than humans alone, they still fall short of the ability to perform exhaustive coverage. Therefore, it is clear that judicious use of limited resources is critical.

Specifically, we assume that our drone collects data that can be accurately interpreted to predict the quantity of interest. For example, we can task a human annotator with identifying tree species from high-resolution drone images, or train a deep learning model to do the same. We further assume that information relevant to this task is contained in remote sensing data, but it is challenging or intractable for a human to label this information directly. This may be because a human annotator does not possess the intuition to label species based solely on low-resolution satellite data. This claim is especially relevant if the remote sensing data contains channels other than Red-Green-Blue, as it is hard for humans to fully interpret this data.

Given these assumptions, the goal is to choose a set of sample locations to observe with the drone. From these observations, we obtain a classification label for each observed pixel in the remote sensing data. We then train a satellite prediction system using these observed pixels as the ground truth training examples. The goal of this

work is to observe regions that serve as useful training samples for this satellite prediction system. This problem is most closely related to prior work in informative path planning (IPP), which was first explored by Binney et al. [12]. This field is concerned with how and where to sample observations with an agent to gain an understanding of phenomena of interest. There are a variety of approaches that are tailored to the objectives and constraints of specific applications. In our domain, an important consideration is that the entire mission must be planned before takeoff because commodity drones do not have the computation or flexibility to re-plan the mission in-flight.

3.3.1 Offline Methods

A number of existing works tackle the problem of offline informative path planning. One class of approaches use ergodic planning, which was introduced by Mathew and Mezic [67]. These approaches seek to optimize an agent's trajectory over an *information map*—which represents how interesting each sample is—subject to the path length and kinematic constraints. Specifically, the goal is to match the spatial time-averaged statistics of the observations with the distribution of the information map. These two quantities are using a Fourier representation. This approach seeks to strike a balance between exploration and exploitation by promoting exploration in some areas with a low information value. A recent extension of this work is sparse-sensing ergodic planning [78], which seeks to optimize the location of a limited set of observations, rather than assuming that observations will be collected along the entire trajectory. Ergodic approaches can be applied in our context, but they only optimize for a spatially-representative set of observations and do not consider other features. Since we assume access to prior imagery, we expect that better performance can be achieved by observing a representative set of these appearance features.

One work that specifically focuses on planning an informative drone flight is TIGRIS [69]. This approach also takes as input an information map, which may be overlaid on a non-flat geometry. It is designed for a fixed-wing drone with a forward-facing camera. Using the parameters of the camera, the planner uses Monte Carlo tree search [14] to build a plan that tries to observe areas of high information without being redundant. Since this approach is designed for target search, it does not prioritize

any notion of diversity but simply tries to observe the most highly-informative areas under the dynamic constraints of the drone.

An approach that is very similar to our work is RIG-Tree [47]. This is a sampling-based approach that uses a branch-and-bound formulation to maintain tractability. One limitation of this work is it does not have an elegant method to deal with a goal state. The algorithm begins by exploring from the start state and will not explore any location that does not leave sufficient budget to reach the goal state. However, it does explicitly encourage that any budget is retained to explore along the path to the goal and rather assumes that the agent will traverse there immediately. We hypothesize that this behavior results in the region around the start being more effectively modeled than the region near the goal.

3.3.2 Online Methods

Significant work has been done under the assumption that the agent can re-plan its trajectory online based on the observations it sees during execution. This requires a sophisticated robotic platform that can sense its environment, process this data into a useful format, re-plan which regions would be useful to explore, and execute this plan. In most cases, the agent maintains some sort of belief of which regions are uncertain, desirable, or a combination of both. At each re-planning iteration, the agent seeks to develop a plan that is likely to optimize its objectives within the operational constraints such as path length.

The work of [75] proposes a generic framework for terrain monitoring with UAVs equipped with a downward-facing camera. This work assumes that the quantity of interest is a scalar field defined over the environment that has spatial correlations, e.g. similar locations will have similar values. The example use-case is modeling the density of weeds in an agricultural application, where it is most important to accurately model regions with a high infestation. Further, it assumes that the UAV is able to re-plan its trajectory in flight, after taking uncertain measurements of the environment. A strength of this work is that it models the altitude-dependent effects of a drone camera: at higher altitudes, the camera can observe more area but the quality of the measurements will be lower. Using an optimization-based framework and a probabilistic sensor model, the algorithm plans a sequence of observations that

decrease the uncertainty while focusing on regions of high predicted weed density. The first part of this path is followed until a new plan based on new observations is developed. This approach is shown to be more effective than a uniform coverage plan at a fixed altitude because it can predict which regions will have more weeds and spend more of the sensing budget there. An extension of this work [96] further explores these concepts with real data. Specifically, they use semantic segmentation to identify weeds in the images and fit empirical models to the segmentation accuracy at different altitudes. This work also shows higher accuracy than a non-adaptive baseline. In both cases, a fundamental strength of the approach is the reason we cannot use it in our domain—online re-planning based on the observed data is critical to the performance increase over the baseline.

A key feature of the previous work is that it seeks to model only regions that were observed with the drone. The main decision is how many times to re-observe a region and at what altitude. In our setting, we wish to make observations with a drone and extrapolate beyond them with satellite data. This goal is somewhat related to that of [87], which seeks to plan a drone flight that collects images to train a deep learning model for land use classification. In this setting, it is assumed that the drone has a downward-facing camera that observes a scene. It also possesses a deep learning model that predicts the land use for each pixel, as well as a measure of uncertainty about the prediction. The goal is to collect images, that when labeled by a human annotator after the mission, can be used to update the model the deep learning model so its performance is better on new data. The authors show that collecting images that the current model is uncertain about leads to better performance after retraining. Therefore, when the drone is collecting data, it should prioritize images that the current model is uncertain about. As the drone explores a region, it generates the uncertainty predictions for each observed image but cannot access the true label. Then, it continuously updates the plan that tries to observe new uncertain images by assuming that unobserved images next to uncertain observed images will also be uncertain. As with the prior work, this approach requires online reasoning to be effective. Furthermore, this approach has a subtly different goal from ours. Their goal is to collect images that can be used to train an accurate model for new drone observations whereas our approach assumes a model exists already for drone data and these predictions can be used to inform a satellite model.

Some prior work formulates the problem in a similar way to our objective. Work by Kodule et. al. [58] and an extension by Candela et. al. [15] explore the problem of where to gather information with a planetary robot, e.g. Mars Rover, given that the whole region has already been observed by a low-fidelity orbiting satellite. Specifically, it is assumed that the world is represented as a grid of pixels each containing a multispectral (8-channel) observation. The goal of this work is to predict hyperspectral data for the entire region using only sparse hyperspectral observations from the agent and the multi-spectral data available everywhere. To accomplish this, the approach uses multiple Gaussian Processes (GPs) [79] that take the spectral values at each pixel as well as the spatial location of each pixel as features. Then the agent uses Monte Carlo Tree Search (MCTS) [14] to plan a set of sampling locations in the environment that decreases the uncertainty of the Gaussian Process. The agent then moves to the next sample on its path, takes a hyperspectral measurement, and replans its future trajectory. This approach is conceptually-similar to our objective, except that it is designed for online reasoning and is computationally demanding due to MCTS.

A similar problem is studied by Edelson et al. [35], where they replace the MCTS planner with an Ergodic planner. While ergodic planning is an offline approach, this work proposes to re-plan an ergodic trajectory online after a number of *in-situ* samples have been collected. After this time, the uncertainty map is updated based on these new observations and a new trajectory is planned. This replanning step allows the agent to collect spectrally-diverse samples. This is because after updating the GP, samples that are similar to previously-observed ones will have low uncertainty and will be de-prioritized during planning.

The goal of this work is to bridge the gap between these two classes of approaches. Specifically, we aim to develop a planner that can function offline while still retaining much of the strong performance of an online planner.

3.4 Summary

Tree detection in drone imagery is a widely-studied topic and many modern approaches use deep learning. Using remote sensing data for the same task has also received significant attention, but largely uses different techniques from classical computer

vision and image processing. There has been work comparing the quality of tree detection from these two modalities, but to the best of our knowledge, no prior studies have compared the performance of the same deep learning approach on both modalities. We propose to conduct this study using DeepForest [106], a commonly-used tree detection method that uses deep learning.

Automatically mapping forest fire fuel has been studied previously using a ground vehicle [25] but this approach only predicts the location of the center of fuel clusters and does not capture its extent or geometry. To the best of our knowledge, no prior work studies building detailed maps of forest fire fuel from the drone perspective. To accomplish this task, we rely on work from the robotics field of semantic mapping. Specifically, we propose to adapt an approach that was developed for an RGB-D sensor [113] to work with a camera and LiDAR and apply this to a multi-sensor drone dataset we collected.

Planning a sparse drone survey for land cover mapping can be accomplished using tools from informative path planning. The work of Rückin et al. [87] studies this application but assumes that the drone can reason online but does not have access to remote sensing data a priori. In contrast, we are interested in using commodity drones that cannot replan online and therefore remote sensing data is critical to inform their flight. The work of Candela et al. [15] addresses a similar problem in the domain of geologic exploration but does provide a robust solution for incorporating goal locations. We leverage ideas from this approach to develop a system that is tailored for land cover classification and carefully considers goal locations.

Chapter 4

Methods

4.1 Datasets

In this work, we use data from a variety of sources. The first is from small, consumer-grade drones that capture only GPS-tagged images. This type of data is relatively easy to collect and representative of what is commonly available to foresters today. The second type of data is from a custom drone payload that records data from multiple sensors at once. This type of rich data has only been collected in limited research settings in forestry. An overview of these datasets can be found in Table 4.1. The third type of data is existing data from remote sensing, specifically an aerial mapping survey that is freely available. Finally, we use annotated forestry data that was captured from a ground vehicle perspective and procedurally generated synthetic data from the same environment. We use this as training data for our system. These datasets are summarized in Table 4.2.

4.1.1 Commodity Drone Data

We use data collected with a DJI Air 2s, a small commodity drone that is commonly used for videography. The data was captured in Stowe, Vermont in the Northeastern United States. The drone was flown in a lawnmower survey pattern at an altitude of 100 meters and the camera was oriented in the downward facing, *nadir*, perspective. This data consists of 225 un-calibrated images that are geo-tagged with commodity-

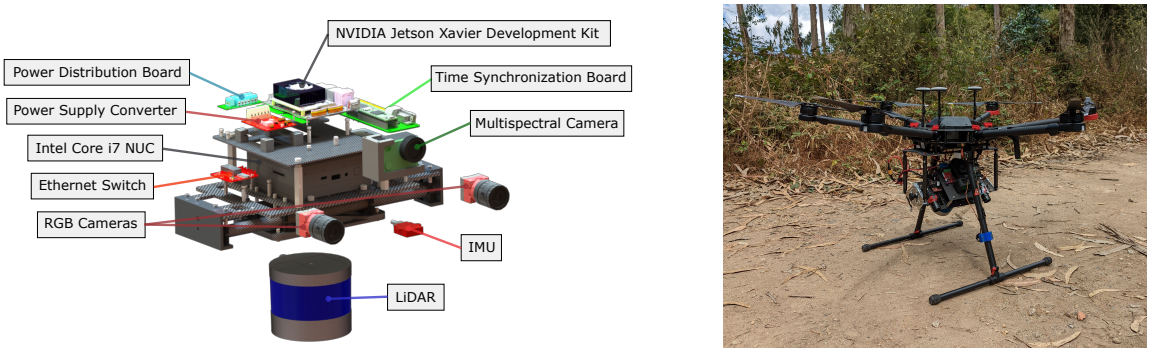


Figure 4.1: On the left is the multi-sensor payload developed by other members of our team. It consists of a LiDAR, stereo RGB camera pair, a multispectral camera, an IMU and a GPS. It also has onboard computation and storage. In most experiments, we used this platform onboard a DJI Matrice 600, pictured right. Photo credits to Winnie Kuang.

grade GPS measurements.

4.1.2 Multi-Sensor Drone Data

Many modern approaches to simultaneous localization (SLAM) and semantic mapping require as input multiple sensing modalities, such as cameras, LiDAR, IMU, and GPS. Therefore, other members of our team built a multi-sensor payload that could be mounted on a drone, as seen in Figure 4.1. This payload is modular and could be mounted to different drones with different inclination angles. In these experiments, we used two large commercially-oriented drones, a DJI Matrice 600 and an AltaX Freefly. We flew a variety of different experiments, both under the canopy and over the canopy. In the under-canopy settings, we flew in small clearings between trees in Portuguese forests under manual control. Babak B. Chehreh, an experienced drone pilot from the University of Coimbra, controlled the drone during these data collections. In these experiments, we surveyed the boundary of the clearing exhaustively by using an oblique payload orientation of 30 degrees from horizontal. This technique was used to collect the *Oporto* and *Coimbra* datasets.

We also conducted a more conventional over-canopy survey using an automated flight planner. We executed a lawnmower coverage pattern over a test site in Pittsburgh, Pennsylvania USA that consisted of trees, shrubs, grasses, and bare earth.

Name	Location	Platform	Environment	Flight Pattern (camera degrees from horizontal)
Stowe	Stowe, VT USA	DJI Air 2s	Forest	Lawnmower (90)
Coimbra	Coimbra, Portugal	Multi-sensor payload (camera only)	Forest path	Manual out and back (30)
Wharton	Hammonton, NJ USA	Multi-sensor payload (camera only)	Forest with road	Manual oval over canopy (60)
Oporto	Oporto, Portugal	Multi-sensor payload	Forest clearing with grass	Manual observations of the boundary (30)
Gascola	Pittsburgh, PA USA	Multi-sensor payload	Trees, shrubs, and grasses	Lawnmower over canopy (75)

Table 4.1: A summary of drone datasets used in this work.

This was conducted at an altitude of 40 meters with the payload facing forward at a slight off-nadir angle of 75 degrees from horizontal. This data is labeled *Gascola* after the name of the area we tested in.

The primary value of this platform was to collect rich multi-sensor data. It also allowed us to roughly replicate the type of data that would have been have been obtained from a commodity drone, such as that described in the previous section. This was done by taking images from a single camera and geo-tagging them with the drone's GPS. We conducted one study where the drone was flown manually in an orbital pattern with an off-nadir payload inclination, so the payload faced outward. This data was collected in the Wharton State Forest in New Jersey, USA, over a coastal pine barren, and is called *Wharton*. Due to a sensor malfunction, this data does not have GPS information.

4.1.3 Forestry Data from the Ground Perspective

Because there is a lack of public annotated forestry datasets from the drone perspective, we used two existing datasets from a ground vehicle perspective. The first consisted of 121 multispectral images of a Portuguese forest collected by Andrada et. al. [5].

These images were manually segmented by the authors into six classes: background, live flammable material (aka fuel), canopies, trunks, humans, and animals. The original paper uses multi-spectral data but we only used the co-registered RGB data that was released. This choice allowed us to deploy the model on the RGB camera onboard our payload, rather than trying to integrate a multispectral camera with similar spectral properties. We refer to this dataset as *Sete Fontes* because it was collected in a region with that name.

We also used a synthetic dataset rendered from a procedurally-generated Portuguese forest as described by Nunes et al. [70]. In this work, the authors procedurally created a forested landscape by first creating the terrain, then placing paths and rocks, and finally adding vegetation. After generating the terrain, they rendered photorealistic images using computer graphics techniques. These renders were created at regular intervals along a simulated ground vehicle trajectory, resulting in 3154 images. They also rendered another image using the same mesh and camera position that described what type of object was observed at each pixel. These classes were slightly more-granular than those used by Andrada et al. [5], but they could easily be aggregated to match the former. There were no examples of humans or animals in this simulated dataset, but these classes were not critical to this work and only occurred infrequently in the *Sete Fontes* dataset.

4.1.4 Optical Remote Sensing Data

In this work, we focus primarily on data collected by the National Aerial Imagery Program (NAIP) multi-spectral data source which contains red, blue, green, and near-infrared bands. This data is collected at an interval of at most every three years over the continental US. The USDA contracts with states to obtain this data from manned aerial surveys. The data is post-processed to provide an ortho-rectified, stitched, and geo-referenced product analogous to satellite imagery. The resolution per pixel is 0.6 meters, which is the highest-resolution freely available multispectral data we are aware of for the US. An example image can be seen in Figure 4.2. The quality of this data has recently been largely superseded by commercial satellites such as Planet Labs [63], but this data is not freely available and therefore not as useful for research purposes.

Figure 4.2: An example NAIP image crop at a resolution of 0.6 meters per pixel. Land cover classes are generally easy to interpret, but small details are lost.

Name	**Location**	**Type**	**Environment**
Sete Fontes [5]	Coimbra, Portugal	Under-canopy images	Forest
Synthetic [70]	Simulation of Portugal	Under-canopy images	Forest
NAIP [103]	Continental US	Aerial imagery	Varied
Chesapeake LULC [20, 82]	Chesapeake Bay, US	Annotated Land Use/ Land Cover segmentation	Varied

Table 4.2: Summary of non-drone datasets used in this work

4.2 Geometric Understanding of Forests using Drone Data

4.2.1 Photogrammetry

We conducted photogrammetry experiments using a variety of datasets. These were collected by both the commodity drone and the custom payload. For the commodity drone, we used all of the data that was collected and did not have to apply any post-processing. This is because the images were only collected after the drone had moved a sufficient distance to avoid redundancy. Additionally, the GPS location where the image was captured was embedded in the metadata.

The data from our custom payload required post-processing to prepare it for the structure from motion software. These images were captured at a high frequency of 10 HZ so consecutive images were often highly redundant. As shown by Young et al. [114], highly redundant images contribute little to the overall quality but we found them to dramatically increase computation times. Therefore, for custom drone payload, we down-sampled the image to 2HZ. If we had GPS data available for the dataset, we tagged each image with the GPS coordinate from the temporally-nearest GPS record.

We found in preliminary experiments that Agisoft Metashape [2], a commercial structure from motion software, consistently produced high-quality results. This software was also used in the work of used by Young et al. [114] for creating models from drone images of conifer forests in the Western US. This work recommends parameters for image alignment and depth map creation, which are the first two steps in the Metashape pipeline. We use these recommended parameters in our experiments. Their work does not provide a recommendation for the later mesh generation and orthomosaic computation steps, so we use the default Metashape parameters which largely prioritize quality over computational time.

4.2.2 Simultanous Localization and Mapping

Since SLAM is not the focus of this work, we choose to use results from our collaborators on these datasets. We use two methods from their experiments, LIO-SAM [93]

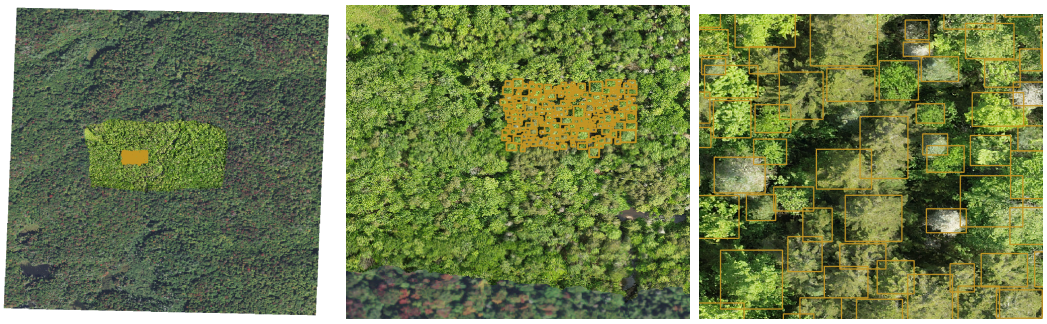


Figure 4.3: The experimental setup with broad-coverage aerial data, medium coverage drone data, and a small set of ground truth tree locations. The same data is visualized at three different scales for clarity.

with the parameters tuned for the forestry domain and a custom SLAM algorithm that combines components of both LIO-SAM and VIL-SLAM [94] to use information from both LiDAR and stereo vision in a tightly-coupled manner. A thorough description of this system can be found in [88]. Both of these approaches provide a continuous estimate of the drone's position and orientation with respect to a static world frame.

4.3 Tree Detection in Top-Down Data

The goal of this work is to study tree detection using multiple types of input data. We consider three sources: aerial imagery, drone imagery, and manual surveys that can provide the location and size of a small patch of trees. The first question we aim to address is the comparative accuracy of tree detections from drone versus aerial imagery using DeepForest [106]. The second is the utility of different site-specific fine-tuning methods compared to simply using the default DeepForest model that is trained on diverse data from the NEON sites across the US. Our goal is to determine whether aerial data such as NAIP is sufficient for detecting trees or whether additional effort must be put in to collect drone imagery and/or field measurements.

For these experiments, we assume that the ground truth trees are fully within the region the drone surveyed and the drone survey be fully within the region covered by satellite data. This is a realistic model for a situation where a forester measured the trees in a region and then flies a drone to survey the surrounding area. The remote sensing data could be taken from any relevant available source, such as NAIP. An

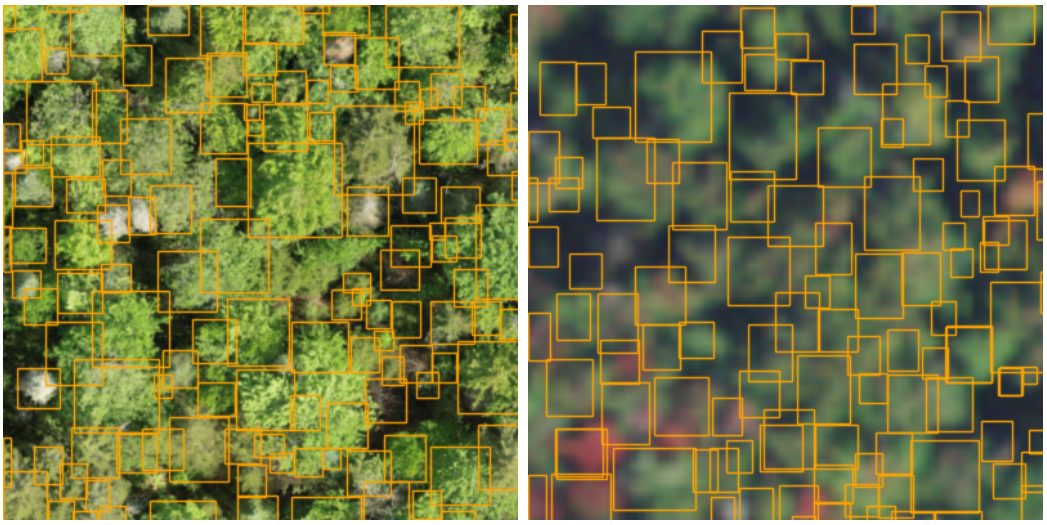


Figure 4.4: Training data from the drone orthomosaic (left) and NAIP (right).

example of the scale of the different types of data can be seen in Figure 4.3.

The first step in this experiment was processing the drone images into an orthomosaic, as described in Section 4.2.1. This orthomosaic had a resolution of approximately 3 centimeters per pixel. Since the drone has GPS, this orthomosaic is roughly registered to a global reference frame. However, there are some errors in the GPS measurements and we noticed a slight mis-registration in both scale and translation. We precisely aligned the orthomosaic to our aerial data using QGis [76]. This provides an interface to select corresponding points between the two datasets. Then, QGis optimizes a translation and scale transform to best align the orthomosaic to the aerial data. We annotated a rectangular region of approximately 950 trees using the orthomosaic data by hand using QGis. Our initial intent was to use field reference data from this site, but this was not available when we were conducting our experiments. We split this data into two roughly-equal regions that are used for two-fold validation.

The overall goal of this work was to assess the performance of a deep learning method for tree detection. We chose to use DeepForest [106] because it is widely used and trained on a comprehensive dataset from the NEON sites [55]. This model is released as a sophisticated software toolbox that handles a variety of pre-processing tasks. Since orthomosaic tiles can be large, both training and inference are performed on crops of the data that can be easily fit into GPU memory. In all the experiments

in this work, we use a crop size of 400px, since this was used for training the original model. For inference, the overlap is 25% to provide additional context to the network. The predictions are filtered using the default confidence threshold of 0.3. After the predictions are generated by the network, non-max suppression (NMS) is applied to each tile to eliminate redundant predictions. Another round of NMS is applied when the predictions from all tiles are aggregated together.

For training, the overlap between tiles is set to 5%, to capture annotations that are on the border between tiles. Examples of these tiles can be seen in Figure 4.4 We train using the default DeepForest settings for the Adam optimizer [57]. When we train on a small set of ground-truth annotations, we use 50 epochs, and when training on a larger set of annotations we scale down the number of epochs inversely to the size of the dataset.

4.4 Semantic Mapping of Forests

The goal of our semantic mapping experiments was to determine the location of forest fire fuel within the environment. Unfortunately, the definition of what is fuel is often vague, such as "anything that can burn is fuel for a fire," according to the US Office of Wildland Fire [37]. Therefore, in our work, we focused on segmenting the environment into four broad classes, canopy, trunks, bare earth, and ground vegetation. Since the goal was to inform an automated ground vehicle, we labeled ground vegetation class *fuel*.

Semantic mapping can be done with a variety of sensing modalities. A relevant work on semantic mapping for forestry, SLOAM [18], focuses solely on detecting tree trunks within the environment and building a 3D map where trunks are represented as cylinders. Because of the distinctive geometry of trunks, they are able to detect them in LiDAR scans. Since we want to distinguish classes that may have similar local geometry, we cannot base our predictions on LiDAR and instead choose to predict semantics using images, as done in the work of Andrada et. al. [5] in the ground vehicle setting. Using images has the added benefit that it is relatively easy for humans to label annotated data for training. This is in contrast to labeling LiDAR data, which takes significant effort as described in SLOAM. Moreover, there are a wide range of semantic segmentation models available that are conceptually

interchangeable.

4.4.1 Semantic Segmentation

To the best of our knowledge, there are no publically available pre-trained models that are useful for our task of vegetation classification in forests using RGB data. Therefore, we needed to train our own models. We conducted two experiments, one on the *Oporto* dataset from Portugal and another on the *Gascola* dataset from the US.

In the Gascola experiments, our objective was to segment the different classes from overhead imagery. Since there weren't any relevant datasets, we chose to label a small amount of data on our own for training. When creating semantic segmentation training data, it is very time-consuming to label the boundaries between each class precisely. This is especially true for natural environments, where different classes are often interlaced at the boundary, such as tree branches over the ground or bare earth transitioning to grass. A relevant work on segmenting ground cover with drones is Davila et. al. [29], where they showed that coarsely labeling regions away from class boundaries was much faster, and the model trained on these coarse annotations still made accurate predictions at the boundaries.

We conducted manual annotations using the VIAME toolkit [30], a free and open-source web annotator. Even though our primary goal was to only distinguish broad classes, we chose to annotate a more granular set of classes loosely inspired by the Anderson13 fuel model [4], a vegetation classification system commonly used in fire modeling. These granular classes, along with their mapping to aggregated classes, can be seen in Table 4.3. Our reasoning for labeling fine-grained classes was that this allowed us more flexibility when it came to future experiments, where we might want to aggregate the classes differently. Furthermore, it gave us the option to train on these classes and aggregate them after prediction. In practice, we found that this additional granularity did not increase the labeling burden significantly, since we rarely had to break up spatial regions that would have originally been considered one coarse class. Rather, we mostly labeled entire regions with more granular designations.

We use a segmentation network based on a transformer architecture called Seg-

Fine-grained class	Coarse-grained classes
Dry Grass	Fuel
Green Grass	Fuel
Dry Shrubs	Fuel
Wood Pieces	Fuel
Litterfall	Fuel
Timber Litter	Fuel
Green Shrubs	Canopy
Canopy	Canopy
Live Trunks	Canopy
Bare Earth	Background
People	Background
Sky	Background
Blurry	Background
Drone	Background
Obstacle	Background

Table 4.3: Classes used in semantic segmentation. These are inspired by relevant classes from the Anderson13 fuel model [4] and also include additional classes relevant to our application.

Former [110]. Given the relatively low amount of real-world images in our training dataset, this network was especially suitable since it showed strong performance on benchmark datasets and good generalization capabilities. We trained this model using the default parameters used in the MMSegmentation [22] implementation.

4.4.2 Semantic Mapping with a Camera and LiDAR

We modified an approach for RGB-D semantic mapping [112] to use LiDAR instead of per-pixel depth data to perform geometric reasoning. First, each RGB image is passed through the best semantic segmentation network from Section 4.4.1 to get a classification result for each pixel. Using the extrinsic parameters relating the LiDAR's orientation and position to the camera, we transformed the LiDAR measurements into the camera's coordinate frame. Then, using the calibrated camera intrinsic, we project each LiDAR point into the image plane. Points within the field of view of the camera are assigned a classification label from the corresponding pixel

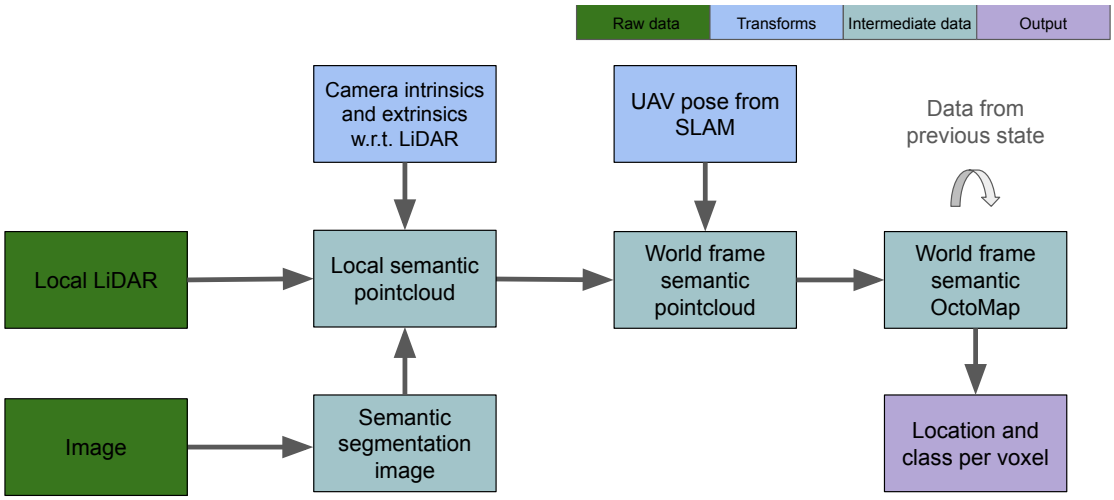


Figure 4.5: An overview of the LiDAR-camera semantic mapping system

in the semantic map. This semantically-textured point cloud is transformed into the inertial reference frame using the current pose of the drone estimated by our SLAM system.

We use an octomap [48] representation to efficiently discretize the generated semantic point cloud into voxels. Each voxel has a resolution of 0.05m and contains information about the predicted classification. Each time a new semantic point cloud is created, it is used to update the global octomap. Since each voxel can contain multiple observations, we use two approaches to determine the aggregate classification. The first method assigns the class label using the highest-confidence prediction from the neural network that corresponds to that voxel. Alternatively, we use a Bayesian method which maintains a probability distribution over the classes. Each new observation is multiplied by the current distribution and then re-normalized. The voxel is then labeled with the most probable class. An overview of the proposed system can be seen in Figure 4.5.

4.5 Informative Path Planning

Automated methods for interpreting remote sensing data are often developed using a sparse set of accurate field measurements. An example of this is the LANDFIRE project, which uses user-submitted plot data about vegetation type to build a predic-

tion model for LandSat data [64]. Drones have the potential to automatically classify vegetation types as described in Section 4.4 and these drone predictions, if accurate enough, could be used to inform the satellite model. A natural question is where to collect sparse drone observations so that they effectively model the surrounding region. Intuitively, the samples should be diverse and focus on examples that are expected to be the most interesting class or most challenging to classify. This intuition is challenging to implement in a domain-flexible way while also respecting operational considerations. A major constraint is that drones have a finite battery life which governs the distance they can cover before returning home to have their battery replaced. This means that the distance of any one flight is bounded. Commodity drones do not expose the ability to algorithmically control the drone in real-time based on the sensor inputs, so the entire trajectory must be planned before the drone takes off. We make some simplifying assumptions to define the type of observations we take. Specifically, we assume that the atomic observation is a plot or a small lawnmower survey of a fixed square size. We further assume that the user specifies a fixed number of plots to visit. Since it will take a fixed amount of time to execute the plot surveys, the maximum available time to traverse between plots is the maximum time of a full flight minus the time taken to complete the surveys. The algorithm's decision variables are where to place these plots and in what order to visit them, subject to the maximum time available to traverse between them.

A good choice of plots depends heavily on the task that is being conducted. However, it's possible that the data may be used for multiple purposes or the method of conducting the task has not been defined when the data is collected. To make our system as general as possible, we assume that in the absence of any other information, collecting a diverse and representative set of samples is desirable. To implement this concept, we need some notion of similarity that is applicable to a variety of domains and input data. We also need a planner that uses this description of diversity to plan a set of observation plots that are diverse and representative.

Feature extraction is the process of converting raw data into a format that captures attributes that might be useful for machine learning. In classical computer vision, feature extraction or "feature engineering" was a widely-studied topic. Many works attribute the success of deep learning to the informative features that are extracted in the early layers of the network. These are optimized for the target problem through

the neural network training process. Since it takes a large amount of labeled data to train these features, it is common to use the first layers of a network trained for one task as a feature extractor or "backbone" for another related task that has less labeled data. However, applicable pre-trained models are not yet ubiquitous for remote sensing, especially given the diversity of modalities. Multiple approaches have been proposed to extract unsupervised features using paired modalities [111] spatial correlations [51] or layer-wise greedy training [84]. In all cases, this still requires training a new network as a feature extractor which can be technologically difficult and computationally intensive.

A recent work called MOSAIKS [83] proposes random convolutional kernels as a strong alternative to pre-trained deep networks for feature extraction. Specifically, they suggest using small crops, e.g. 3x3, from the dataset as convolutional filters and then applying a non-linear activation. This process is extremely computationally efficient and notably requires no neural network pre-training. The authors show that these features are remarkably good for a variety of predictions on geospatial data. Specifically, they are better than using a CNN pre-trained on a different task as a feature extractor, and almost as good as a CNN trained for the task in question.

Because of the strength and flexibility of this approach, we use it as the basis for our feature extraction method. The original work uses 1024 kernels to extract features, so the features consist of significantly more data than the input. This is because the convolutional features are produced at the same resolution as the input image. In their work, the authors address this issue by spatially averaging the data across large cells. However, in our work, we require features that capture variation on a local level. Therefore, we retain the spatial resolution but reduce the number of features with dimensionality reduction. This technique was employed in a related informative path planning work by Candela et al. [15], except the input was hyperspectral data rather than the convolutional feature maps. In both cases, the features are highly correlated with each other, so much of the information can be represented by a significantly smaller number of features. We use Principal Component Analysis (PCA) [53], a widely used statistical technique for reducing the dimensionality by finding the linear projection that retains the most variance in the original data. We use the first 6 principal components as features. This number of features was chosen because preliminary classification experiments showed

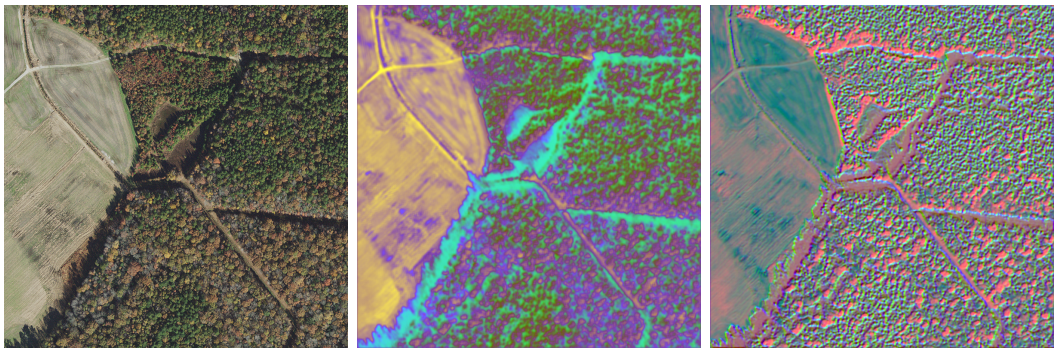


Figure 4.6: Example unsupervised features generated by MOSAIKS and PCA. The first image is the input data and the next two images are the six features, visualized as the channels of two RGB images. For visualization, the data is centered at 0.5 and clipped at the third standard deviation.

performance plateaued after this number. A useful property of PCA is that the features it produces are uncorrelated. To make our feature representation even more consistent, we standardize each feature to have zero mean and unit variance. An example of feature extraction can be seen in Figure 4.6. This shows that different types of land cover have different feature representations.

4.5.1 Gaussian Process Uncertainty Modeling

The goal of extracting meaningful features is to enable uncertainty modeling. Gaussian Processes (GPs) [79] are a principled tool for quantifying prediction uncertainty. They are a kernel-based method, where the kernel defines the similarity between two features. This can be fit to data or set using expert knowledge. Because of this, they are used as a key component of works on sensor placement [60] informative path planning [15, 16, 36]. An important property of GPs uncertainty is that it only depends on the features and not the associated values. This makes them applicable to offline planning. In contrast, uncertainty estimation approaches built on ensembles of prediction models require the label of a proposed sample to compute the uncertainty reduction.

4.5.2 Long Horizon Informative Path Planning

To plan a path, we need an algorithm to plan where to take observations that can effectively reduce the uncertainty of the entire map. We make two observations that

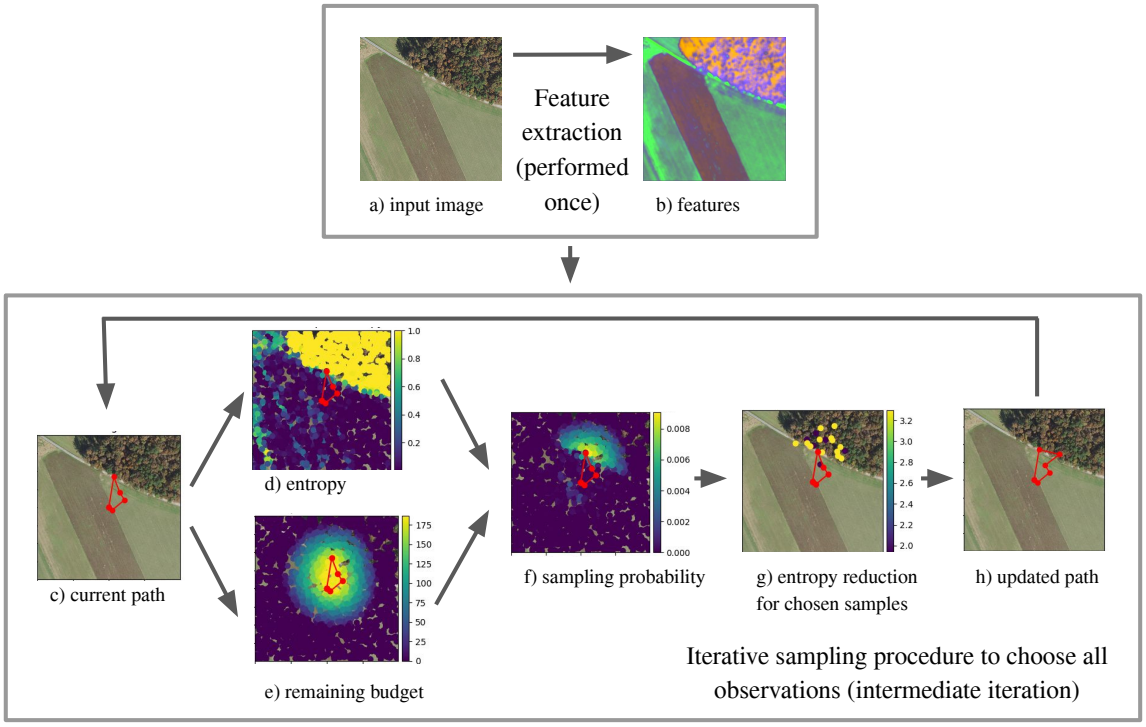


Figure 4.7: This figure describes the workflow of a generalizable long-horizon path planner for selecting a set of drone observations. As a first step, unsupervised features are extracted from the image. Then, samples are added iteratively to the path to reduce the entropy of the map while respecting the path budget.

inform this work. First, it is assumed that we can fly in any direction and there are no obstacles since we are flying above the canopy. Second, the drone has to stop to execute each survey, so kinematic considerations are effectively removed. Given the scale of distances between points and the drone's rapid acceleration, the time to traverse between points is assumed to be proportional to the distance between them.

We employ a sampling-based strategy to build a long-horizon offline path. This takes in an initial location, a number of samples, and a traversal budget. A visualization of the proposed remote-sensing aware planning through observation random sampling (RAPTORS) can be seen in Figure 4.7. At each iteration, the current path is provided as input. Then the uncertainty for a set of candidate locations is computed using a GP and the remaining flight budget taking into account the current path is computed. Then the uncertainty and remaining budget are multiplied and normalized to obtain a probability of evaluating each sample further. A set of samples

are selected and the entropy reduction is calculated if each one were added to the GP model. Then the best sample is added and the path ordering is recomputed using a traveling salesman solver. We use the PythonTSP [40] implementation of Simulated Annealing [72]. Then the process is repeated to plan the next observation until the requested number of observations are planned for. Only then is the path executed by the drone. In addition to the version of the algorithm that weights all samples equally, we propose RAPTORS_rare to prioritize discovering rare classes. If a classification prediction is available, each sample is weighted based on the inverse frequency of that class in the region. This up-weights the samples that are predicted to be from rare classes so more examples will be included in the survey. An implementation of this approach can be found here.

4.6 Summary

We use a variety of types of datasets including images from a commodity drone, multi-sensor data from a custom payload, real and simulated data from the ground perspective, and imagery from crewed aircraft. To extract geometric information from the commodity drone data, we use SfM. Because the custom payload has multiple informative sensors, it can be processed online with SLAM or offline with SfM. The experiments will analyze the quality of SfM on a variety of scenes and present a comparison between SfM and SLAM.

We are interested in comparing the performance of tree detection using the commodity drone data processed with SfM compared to the same approach with aerial data. The drone data is high quality, but our experiments aim to study whether the readily-available aerial data can accomplish the same task.

We present an approach for fuel mapping using semantic segmentation and LiDAR-based SLAM. Because there is limited ground truth data for the semantic mapping task, we focus our quantitative evaluation efforts on assessing the performance of semantic segmentation in this domain.

Finally, we show how remote sensing data can be featurized using convolutional kernels taken from the dataset and compressed with PCA. We develop a path planning approach that seeks to select representative samples from the environment using this representation. Our experiments address whether the samples proposed by this

planner are more useful for training a remote sensing classifier than those from a coverage planner.

Chapter 5

Results

5.1 Photogrametry on Drone Forest Images

An important first step in multiple approaches we used is understanding the structure of the forest scene. As one solution to this problem, we ran Agisoft Metashape with the parameters from Young et al. [114] on a variety of datasets. Three datasets shown in Figure 5.1 are representative of the types of data we experimented with. The first dataset, *Stowe*, was collected with a lawnmower survey from a commodity drone. The locations of all the cameras were properly estimated and the results look good throughout. Some geometry of the trees can be seen, but much of the fine detail is lost, which is common for photogrammetry. The next dataset is *Warton*, which was collected with the custom payload at an off-nadir angle of 60 degrees from horizontal and manual flight pattern observing roughly the same region from different angles. All of the cameras were properly aligned, but the quality of the mesh varies dramatically. In the center where there are multiple observations, the results are fairly good. However, both the structure and texture become blurrier farther away from the central area. This is likely because there are fewer camera views to triangulate and those regions are farther away. This mesh is darker than the *Stowe* it was underexposed, due to a fixed exposure between collects. It is important to balance under- and over-exposure, especially in regions with different brightness. A further consideration is that longer exposures are more prone to motion blur, which is especially common while the drone is turning. Finally, *Coimbra* is a dataset where

the drone follows a path between trees. The data is captured with the custom payload facing forward at 30 degrees from horizontal. A large portion of this trajectory appeared to be roughly accurate. However, in multiple instances, a set of cameras and the associated mesh were rotated relative to the correct orientation. This resulted in odd artifacts that made the mesh effectively unusable. Furthermore, the level of visual detail was fairly low.

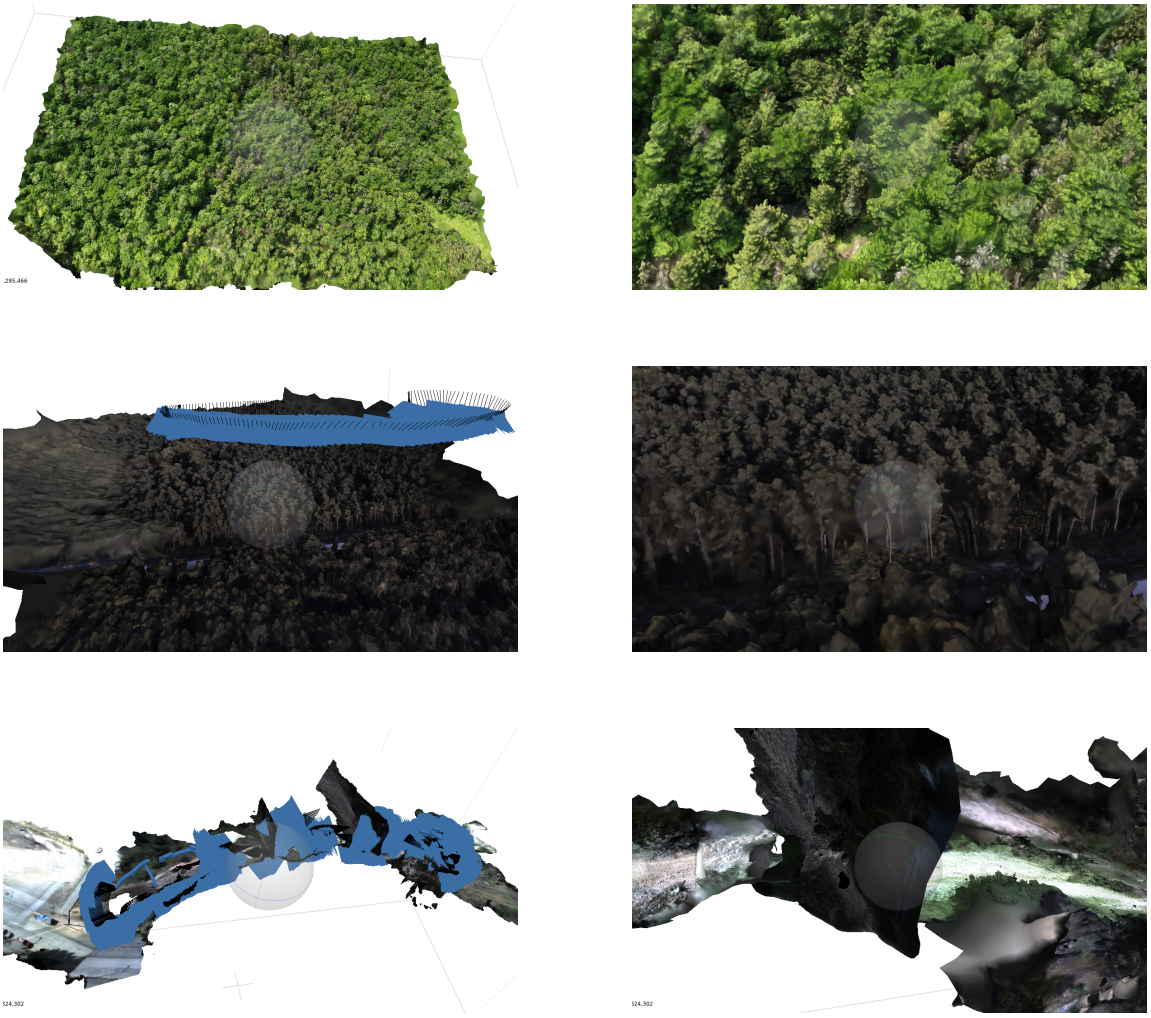


Figure 5.1: 3D reconstructions using Agisoft Metashape on three different environments. From top to bottom, they are a lawnmower pattern with a commodity drone, The full map is shown to the left and a zoomed-in inset is shown to the right.

These results support the common practice of automated drone surveys using

a lawnmower pattern since this method yielded consistently high-quality results across all the datasets we experimented on. However, it also shows that manual flight patterns over the canopy yield some useful data and challenging under-canopy conditions are somewhat successful. This suggests that traditional drone surveys are well-motivated, but further research on photogrammetry for more challenging forest scenarios may yield useful results.

5.2 Simultanous Localization and Mapping in Forest Environments

We were interested in comparing the results of SLAM to photogrammetry because these two systems represent an interesting trade-off. SLAM approaches generally require multiple synchronized and calibrated sensors, with LiDAR being a common requirement. However, they can run in an online manner which can be used to inform robots' next actions. Photogrammetry requires only images, with optional GPS. The downside is it is computationally expensive and must be run in a batch after all the images have been collected.

We conduct this comparison on the *Gascola* dataset that was surveyed with lawnmower coverage and the custom payload at a slight off-nadir angle. We ran photogrammetry using the default parameters. Francisco Yandun ran a tuned version of LIO-SAM [93] and conducted a comparison between the two approaches.

As shown in Figure 5.2, the two maps agree fairly well in most regions. A notable source of high error is the tops of trees. In general, our qualitative assessment suggested that the SLAM approach was better at capturing these fine details compared to photogrammetry. This is because it had the benefit of explicit 3D data from LiDAR, while the photogrammetry approach had to infer 3D information from images. This often leads to sharp points and fine detail being lost. This shows that in offline applications of over-canopy mapping, photogrammetry is likely sufficient.

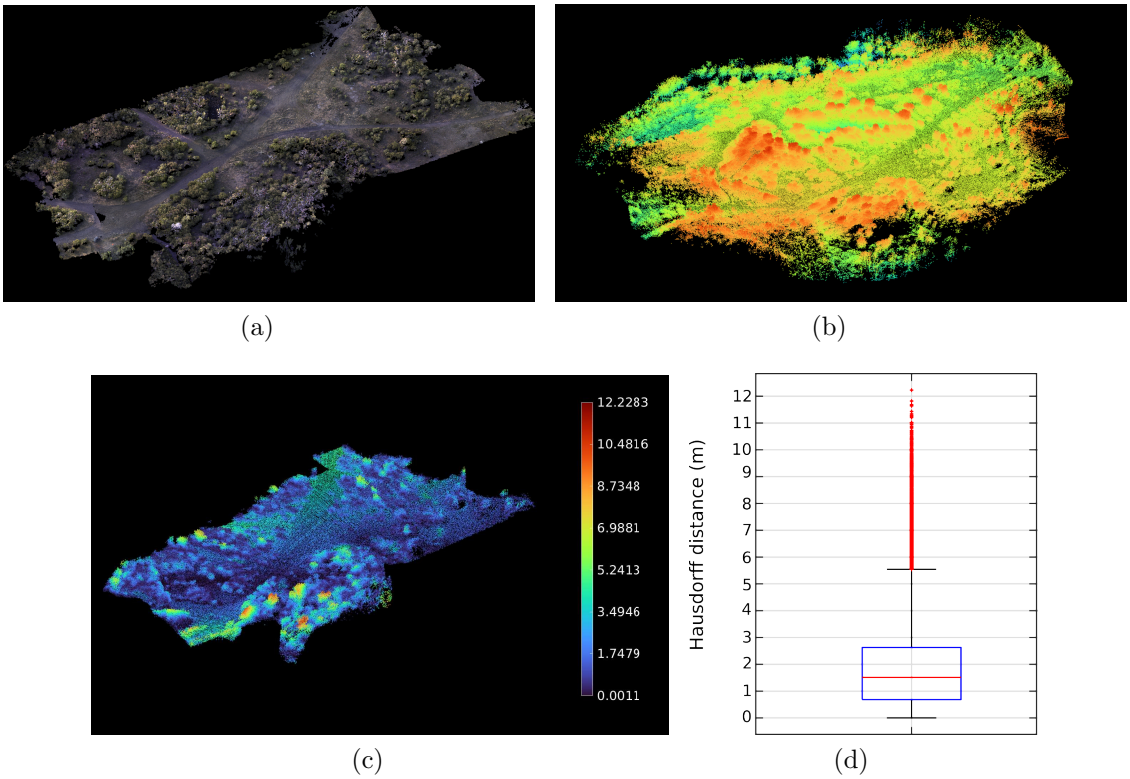


Figure 5.2: Point cloud maps of the A) photogrammetry baseline, B) SLAM outcome. The two maps were compared using the Hausdorff distance, whose result is visualized as C) the SLAM map colored according to this metric and D) a boxplot showing the error distribution. Note that the baseline and the SLAM clouds colormaps correspond to RGB and height values, respectively. This analysis was conducted by Francisco Yandun and this figure appeared in [7].

5.3 Tree Detection using Data at Multiple Scales

The goal of these experiments is to provide insight into the performance of DeepForest in different settings, with the eventual goal of informing what data would be sufficient for useful tree detection.

The first aspect we explore is the influence of the image resolution that is provided to the detection network. The authors of DeepForest state that this parameter is important, but to the best of our knowledge do not provide any experimental analysis of its impact. The first situation this is useful in is when high-resolution data is available. Then, these experiments can inform what resolution to downsample to for best performance. The second is when planning drone data collection or conducting post-processing. As long as the resolution is suitable for photogrammetry, there is no reason to collect data that is higher resolution than is needed for network inference. Similarly, there is no reason to export an orthomosaic that is higher resolution than needed, since this can often result in very large images.

To study this phenomenon, we use data from both the drone orthomosaic and NAIP. For each type of data, we conducted experiments with both the DeepForest model trained on NEON data, termed *pretrained* and a model that was finetuned on one half of the data from the site, termed *finetuned*. In both experiments, we trained the network for 50 epochs using the default settings of the Adam [] optimizer provided by DeepForest. The confidence threshold at inference was set to 0.3, which is also the recommended default. The predictions are evaluated on the test set, which is spatially distinct from the training set. The experiment is repeated with 2-fold validation, using one half as the training set and the other half as the test set. To provide increased confidence in the results, each configuration is repeated three times. These six trials are repeated for 20 different inference resolutions sampled logarithmically from 0.01 to 0.7 meters per pixel. In the *finetuned* experiments, the training resolution is the same as the inference resolution.

A common metric to assess the quality of a detection prediction is intersection over union (IoU). This is the ratio of the area of the overlap between the prediction and groundtruth to the total area covered by both. One metric we use to assess the quality of the predictions is the mean intersection over union (mIoU), which is the IoU with the closest groundtruth tree, averaged over all predictions. We also

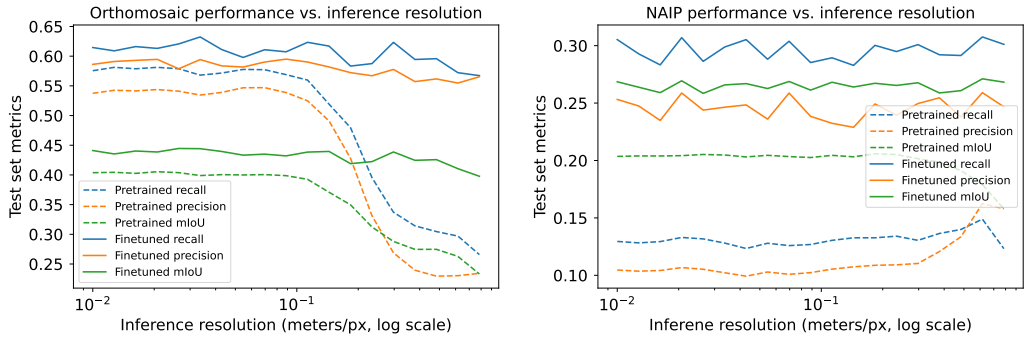


Figure 5.3: Tree detection metrics for drone data and NAIP versus inference resolution.

compute precision and recall. The authors of DeepForest state that a predicted tree with an intersection-over-union (IoU) of 0.4 with a groundtruth tree, is useful for ecological applications. A true positive *TP* is when a predicted tree has sufficient overlap with a ground truth tree. A false positive *FP* is a spurious detection when a tree is predicted but does not have sufficient overlap with any ground truth tree. Finally, a false negative *FN* is a missed detection where no prediction overlaps with a given ground truth tree. Precision represents the fraction of predictions that match a ground truth and recall represents the fraction of groundtruth trees that are matched by a prediction. These equations are summarized below:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$
$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

The results of these experiments are presented in Figure 5.3. The pretrained model on the drone data exhibits consistent performance across all metrics between 0.01 and 0.1 meters per pixel. As the resolution becomes coarser than 0.1 meters, the performance quickly deteriorates. With the finetuned model, performance remains much more consistent across resolutions from 0.01 to 0.7 meters per pixel. This suggests that low-resolution data is too far out of distribution for the pretrained network, but the finetuning allows it to still perform well even on data that is different from what was seen during the NEON training. The NAIP data shows a different trend. For both the pretrained and finetuned models, the performance is consistent

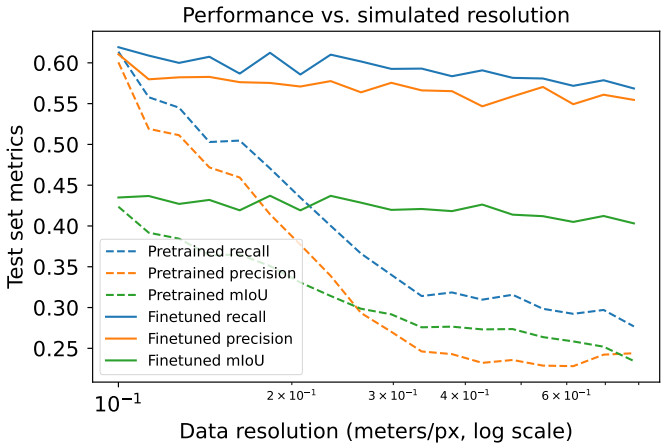


Figure 5.4: Performance of tree detection with downsampled drone data, used to simulate remote sensing data of different resolutions.

but low for all resolutions. The exception is for the pretrained model, the mIoU drops, and the precision increases for the coarsest data.

Because of the poor performance of NAIP data, we wanted to explore other options for remote sensing data. Because it is challenging to obtain higher-quality, commercial remote sensing data, we instead chose to simulate varying resolutions of data by downsampling the orthomosaic to simulate lower-resolution remote sensing products. This has the added benefit of removing confounding effects from different data products and focusing solely on resolution. Given our previous experiments, we chose to perform inference at a resolution of 0.1 meters per pixel. Therefore, we simulated data that was lower resolution than this threshold by downsampling the orthomosaic to resolutions between 0.1 and 0.8 meters per pixel, chosen logarithmically.

The results of this experiment are presented in Figure 5.4. For an input resolution of 0.1 meters per pixel, the performance of the pretrained and finetuned models are similar. As the resolution decreases, the performance of the pretrained model drops dramatically. At approximately 0.3 meters per pixel, the performance of the pretrained model begins to decrease less rapidly. In contrast, the performance of the finetuned model only decreases slightly as the resolution drops. At a resolution of 0.6, the performance of the downsampled orthomosaic is still significantly better than that of the NAIP data. This suggests that despite the comparable resolution,

something about the orthomosaic is more suitable for tree detection with DeepForest. One potential explanation is the difference in colors between the two types of data.

Finally, we propose a set of experiments to explore different approaches to using multiple types of data. In all settings, the training and inference resolutions are 0.1 meters. Each setting is run using two-fold validation with three trials per fold. These experiments are summarized below:

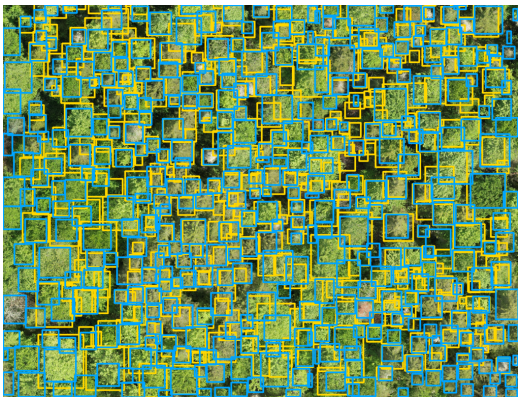
- **Pretrained ortho:** This is applying the DeepForest model trained on NEON data to the drone orthomosaic. To deploy this approach, only drone data is needed.
- **Finetuned ortho:** This is fine-tuning the DeepForest model trained on NEON data with a small amount of site-specific data. This requires both drone data and labels from field work or manual annotations.
- **Pretrained NAIP:** This is applying the DeepForest model trained on NEON data to NAIP. This requires only NAIP data.
- **Finetuned NAIP:** This is fine-tuning the DeepForest model pretrained on NEON data with a small amount of site-specific data. Similar to *finetuned ortho*, this requires two types of data, NAIP and a small ammount of tree labels.
- **Finetuned NAIP on predictions from pretrained ortho (Finetuned NAIP[pretrained ortho]):** This is a model for NAIP data. Instead of finetuning this model on labels taken from field work or manual annotations, this model is finetuned using trees predicted by the *pretrained ortho* model using drone data. This experiment does not require labeled data and only needs drone and NAIP data. The hypothesis is that the increased number of training samples can compensate for the decreased quality obtained by training on automated predictions. Furthermore, it removes the need for manual labeling.
- **Finetuned NAIP on predictions from finetuned ortho (Finetuned NAIP[finetuned ortho]):** This is also a model for NAIP data. Similar to the previous experiment, the model for NAIP is finetuned using predictions on drone data. In this case, the predictions are generated using the *finetuned ortho* model. This means that all three types of data are required: labeled data, drone data, and NAIP.

Experiment	Recall	Precision	mIoU
Pretrained ortho	0.614 ± 0.000	0.601 ± 0.076	0.424 ± 0.007
Finetuned ortho	$\mathbf{0.620} \pm 0.026$	$\mathbf{0.608} \pm 0.079$	0.440 ± 0.021
Pretrained NAIP	0.125 ± 0.004	0.100 ± 0.009	0.202 ± 0.004
Finetuned NAIP	0.292 ± 0.030	0.242 ± 0.017	0.261 ± 0.018
Finetuned NAIP[pretrained ortho]	0.326 ± 0.052	0.289 ± 0.034	0.279 ± 0.026
Finetuned NAIP[finetuned ortho]	0.319 ± 0.051	0.316 ± 0.025	0.262 ± 0.029
Pretrained DS ortho	0.303 ± 0.010	0.231 ± 0.033	0.266 ± 0.014
Finetuned DS ortho	0.576 ± 0.034	0.563 ± 0.068	0.412 ± 0.033
Finetuned DS ortho[pretrained ortho]	0.612 ± 0.015	0.563 ± 0.076	0.420 ± 0.013
Finetuned DS ortho[finetuned ortho]	0.615 ± 0.024	0.546 ± 0.076	$\mathbf{0.444} \pm 0.025$

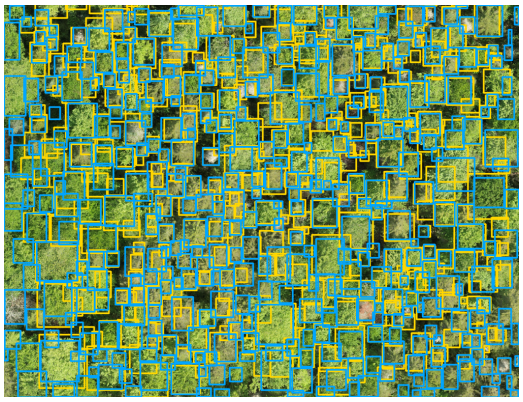
Table 5.1: Tree detection results using multiple experimental strategies. The approaches are evaluated on precision and recall at an IoU threshold of 0.4 and the average IoU for all predictions.

- **Downsampled orthomosaic experiments:** As shown in the initial experiments, NAIP data shows poor performance for tree detection. However, the drone orthomosaic downsampled to the same 0.6 meters per pixel resolution shows better performance. Therefore, we repeat all four NAIP experiments with the downsampled orthomosaic and these are notated as *DS ortho*.

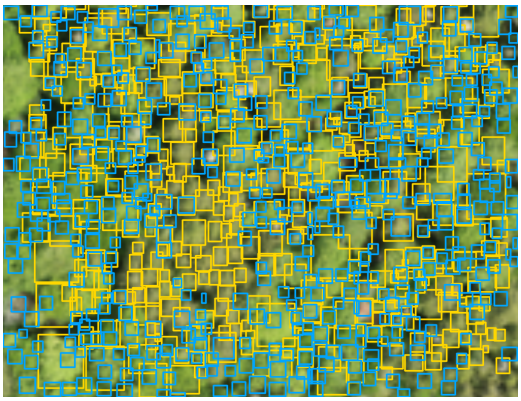
Qualitative results from a subset of the experiments can be seen in Figure 5.5. This shows that both pretrained and finetuned models for drone data produce high-quality predictions. The pretrained model applied to either the downsampled orthomosaic or the NAIP data yields poor performance, with NAIP being especially bad. In both cases, the model predicts a sparse set of detections that are smaller than the real trees. Finetuning produces significantly better performance than the pretrained model for both types of data. In Table 5.1 we summarize the quantitative results of all these experiments. As seen here, the best performance for both recall and precision is obtained by the model finetuned for the drone orthomosaic. This is expected because this data is the intended use-case for DeepForest and the fine-tuning allows the model to be adapted for the specific site. The predictions on NAIP are much lower quality than those from the orthomosaic, even when fine-tuning is used. This suggests that NAIP is a challenging dataset to use and has significantly different properties than the orthomosaics used to train DeepForest. Without finetuning, the downsampled



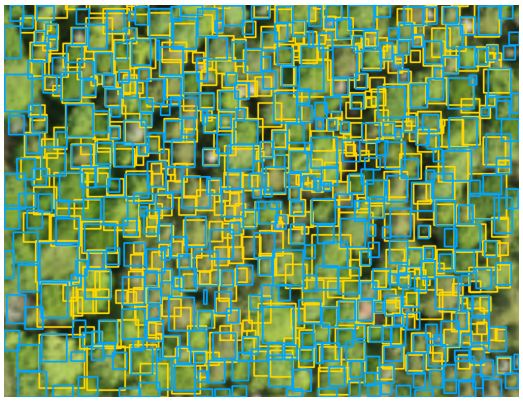
(a) Pretrained ortho



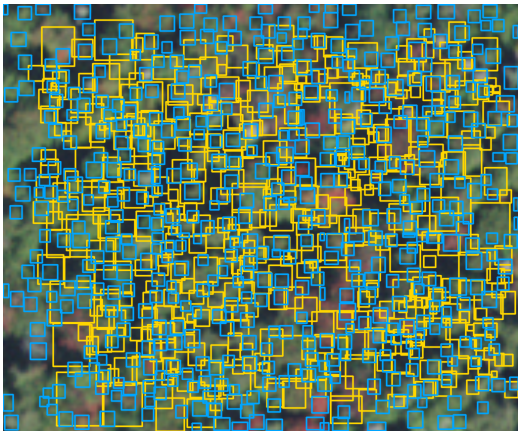
(b) Finetuned ortho



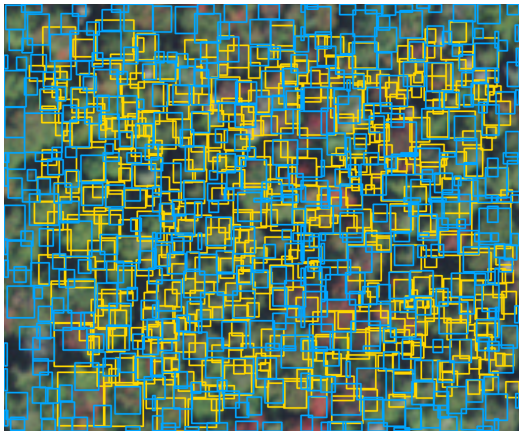
(c) Pretrained downsampled ortho



(d) Finetuned downsampled ortho



(e) Pretrained NAIP



(f) Finetuned NAIP

Figure 5.5: Tree detections on the test set for drone orthomosaic, downsampled orthomosaic, and NAIP data. The left column is the pretrained model while the left is finetuned with site-specfic data. Predictions are in blue and groundtruth is in gold. Best viewed zoomed in.

orthomosaic yields poor performance. However, with finetuning the performance approaches that of the original resolution orthomosaic. This is interesting because this downsampled data is the same resolution as NAIP but has much better performance. A similar trend can be seen for both the NAIP and downsampled orthomosaic data. The worst performance is seen in the pretrained model and finetuning on the small set of ground truth annotations yields significantly better performance. However, the best performance is achived by training on the predictions from the drone data. This is promising because it suggests that in some contexts a model for remote sensing data can be trained using only predictions from the drone.

5.4 Semantic Mapping for Vegetation Classification

The goal of this work is to build a 3D map of the environment that is annotated with the type of vegetation at each location. A challenge of this work is that we did not have access to field-reference data for the true locations of the different types of vegetation. Therefore, we conduct much of our quantitative evaluation on the semantic segmentation predictions, which can be more easily compared to human-labeled images.

5.4.1 Image Segmentation

We conduct two experiments related to semantic segmentation. The first is a study using under-canopy data to determine the potential for synthetic pretraining and the impact of the number of training images on performance. The second is an evaluation of the model we used in over-canopy mapping with a small set of training examples.

Sete Fonte Experiment

Given the limited availability of real data and the labor-intensive nature of labeling to obtain ground truth, we explored the utility of models trained with simulated (*synthetic*) data. We conducted three types of experiments: models trained solely with *synthetic* data, models trained with real data (*Setes Fontes*), and a mixture of

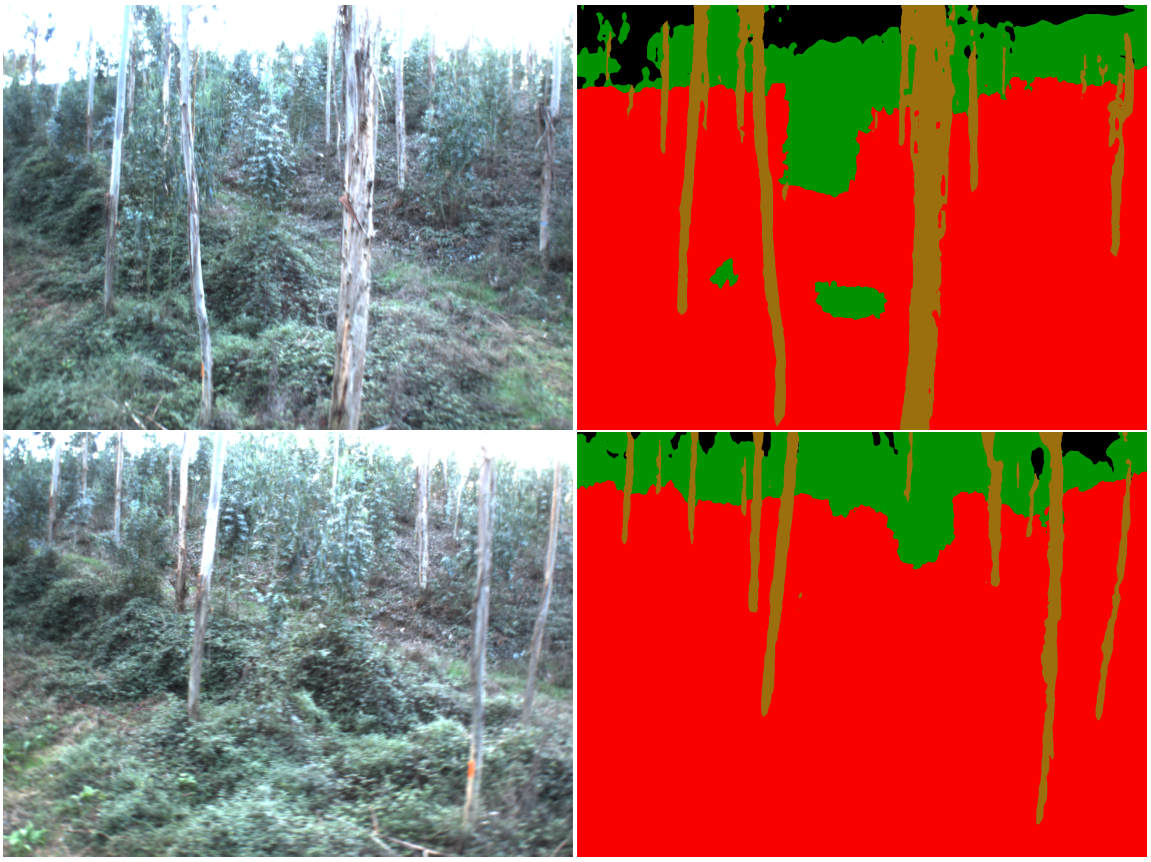


Figure 5.6: Predictions on the *Oporto* dataset. Black is background, red is fuel, brown is trunks, and green is canopy.

both. For the last two cases, we trained different models with a variable number of real images to evaluate the performance of the model as the number of real labeled images varied. Thus, we conducted training experiments using (or adding) 7, 15, 21, 30, 60, 91, and 121 data points from the *Setes Fontes* dataset. We trained for 10000 iterations and evaluated each model on 30 *Setes Fontes* images not seen in the largest training split. We replicate this experiment over five folds of the data. For all three models, the base networks were first pretrained with the *CityScapes* dataset [23]. The mean Intersection over Union on the test set of each of these configurations is shown in Figure 5.7.

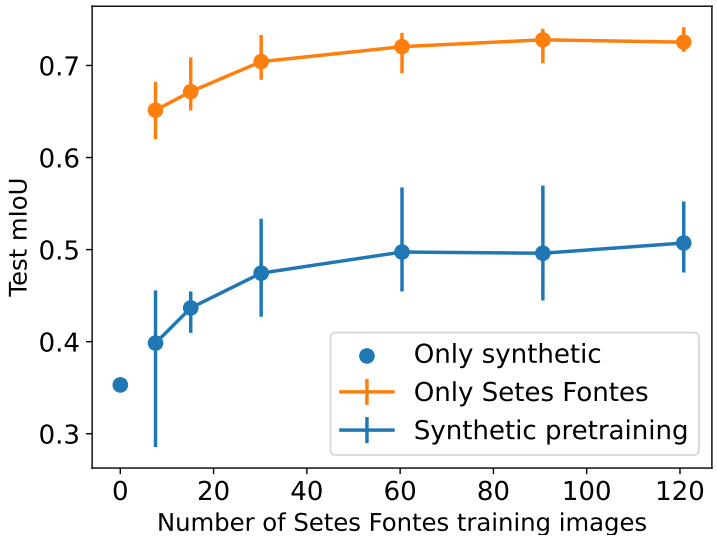


Figure 5.7: Test mIoU for very few training images on the *Setes Fontes* dataset. Error bars represent minimum and maximum results across the five folds of *Setes Fontes*.

It is interesting to note the relatively high performance of a model that used only 7 real images. Also, the model trained solely on synthetic data fails to generalize to real data, even after properly accounting for differences in mean and variance of both datasets. We found that combined real and synthetic data performs worse than training the same model only using real data. This suggests that the synthetic data comes from a completely different distribution than the real one, making its contribution detrimental. In the future, we plan to keep researching the causes of this interesting outcome. It is possible that simple attributes such as saturation or

spatial resolution is the cause of this domain shift. Alternatively, it's possible that significantly more effort needs to be put into realistic simulations for them to be useful to train prediction models.

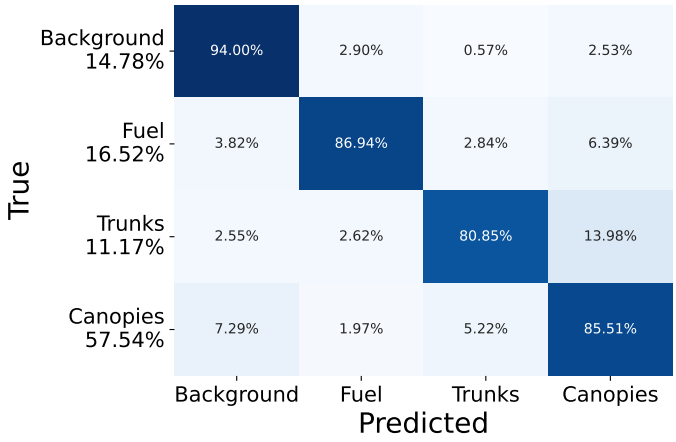


Figure 5.8: Confusion matrix for the *Setes Fontes* test datasets normalized per class with the true fraction of each class reported on the y axis labels.

Since the model trained in 121 images (80% of the *Setes Fontes* dataset) showed the best performance in these experiments, we conducted further experiments on it. The confusion matrix on the *Sete Fontes* test set is shown in Figure 5.8. This shows that all four classes are predicted with reasonable accuracy. A common source of confusion is between trunks and canopies, which is understandable because they frequently overlap. Canopies are also confused for background, which includes the sky. Even though these two classes look very different, the sky can often be seen through the canopy, and the network miss-classifies these fine details. In most cases, this error is harmless because no LiDAR information should correspond to the sky pixels. Finally, the most common error for fuel is canopies, which is understandable because they are both leafy vegetation. Information about the height, either provided to the model or used in post-processing, could help disambiguate this confusion.

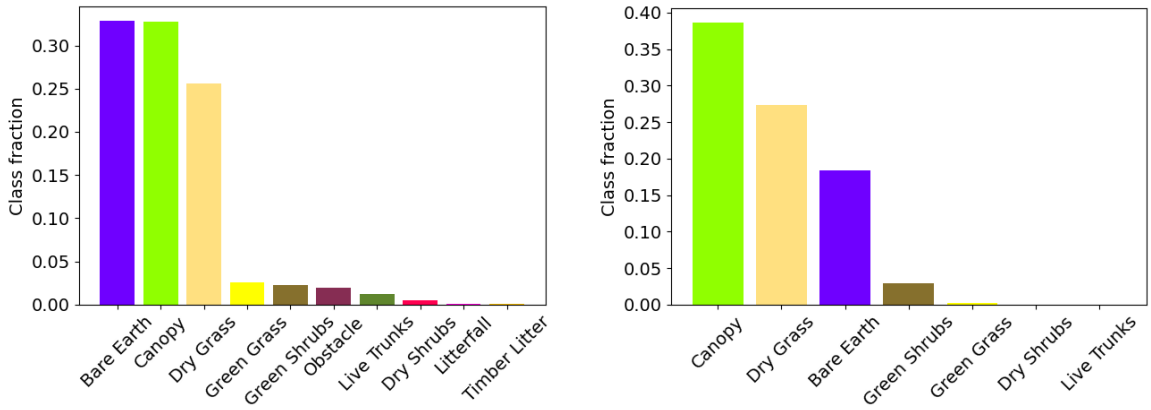
Since we are mainly interested in the fuel instances, we aggregated the background, trunks, and canopies in a single non-fuel class. In that case, we obtained an IoU of 78.2% and 95.3% for `Fuel` and `Not Fuel`, respectively, which yields a mIoU of 86.7%. This shows that our system performs well at its primary task of identifying fuel.

The end goal of this model is to be useful on the semantic mapping task on

the *Oporto* dataset, which wasn't seen during training. We show two qualitative examples of predictions in Figure 5.6. The predictions still appear fairly accurate, despite the change in camera perspective from ground to aerial and different image characteristics.

Gascola Experiment

In this experiment, we used two different datasets from *Gascola* which partially covered the same region. We train on one dataset and test the model on the other. Because of the small number of images that we used in this experiment, these two datasets had different class distributions as shown in Figure 5.9, though they had the same three main classes. Each of these three classes corresponded to a different aggregate class (Background, Canopy, and Fuel), so it was important to effectively tell them apart.



(a) (b)

Figure 5.9: This shows the fraction of pixels per class for A) the train set and B) the test set. Note that the three dominant classes Canopy, Bare Earth, and Dry Grass are common across both collections but the comparative frequencies are somewhat different. These correspond to our aggregate Canopy, Background, and Fuel classes respectively. The fractions of other classes are fairly small, and some from the training set are entirely absent in the test set.

The quality of predictions is shown in Table 5.2. We summarized the IoU, precision, and recall for each class. The performance is fairly good on the most common classes,

but, as expected drops on the rarer classes. As seen by the qualitative examples in Figure 5.10, there are instances where the predictions on the granular classes are incorrect, but the aggregate class is correct.

Class	IoU	Precision	Recall
Canopy	70.05	84.77	80.13
Dry Grass	79.7	93.75	84.17
Bare Earth	78.53	88.12	87.83
Green Shrubs	3.27	21.72	3.71
Green Grass	0.0	0.0	0.0
Dry Shrubs	0.0	0.0	0.0
Live Trunks	0.05	0.05	84.09

Table 5.2: Evaluation results of the SegNext [42] network with the Anderson Fuel Model [4] for semantic segmentation in a forestry environment.

5.4.2 Projecting Segmentation into 3D

The final goal of semantic mapping is to develop a model of the environment and what class different regions are. To evaluate the feasibility of this, we conduct two experiments using multi-sensor data.

The first experiment is conducted on the *Oporto* dataset, which is collected in a clearing in Portugal with the custom payload at an orientation of 30 degrees from horizontal. For semantic segmentation, we use the model trained on *Sete Fontes*. For localization, we use the vision-LiDAR SLAM system from [88]. The results are shown in Figure 5.11. This shows that the system was able to determine that there was significant fuel at ground level and correctly identify the tree trunks in the environment. The latter is important because trunks could be used as a localization aid in a similar way to SLOAM [18]. One issue that we observed was that multiple scans were not perfectly registered with each other. This is in contrast to the final point cloud derived from SLAM, where fine details are captured precisely. Our hypothesis is that because the SLAM system uses a pose-graph [31] formulation, information from loop closures can be used to update the historical pose to make it more accurate. However, the semantic mapping system only uses the most recent

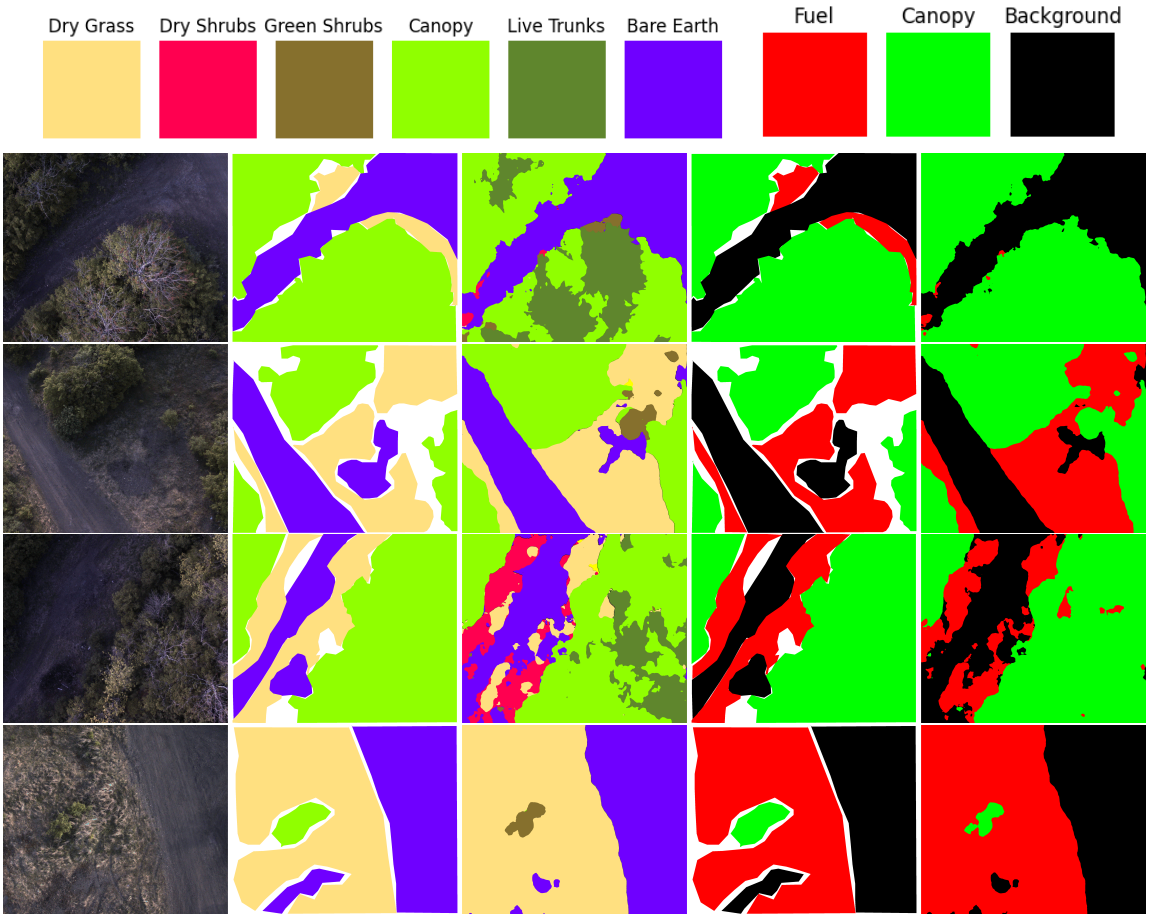


Figure 5.10: Qualitative semantic mapping results from the test set. The results are shown both for the predicted classes and the aggregated ones, with colors visualized in the top rows. White regions in the ground truth represent areas that were ambiguous to the human annotator. Overall the predictions match the ground truth well and boundaries are well-defined.

SLAM pose estimate, which is likely to be less accurate than the final optimized version.

A limitation of the first experiment was our lack of ground truth data to compare against. In our second experiment on the *Gascola* dataset, we still did not have field reference data but tried to approximate this as accurately as possible. We chose to label an orthomosaic derived from a 3D model by hand. We used QGis [76], an open-source software for interacting with geospatial data. We labeled three

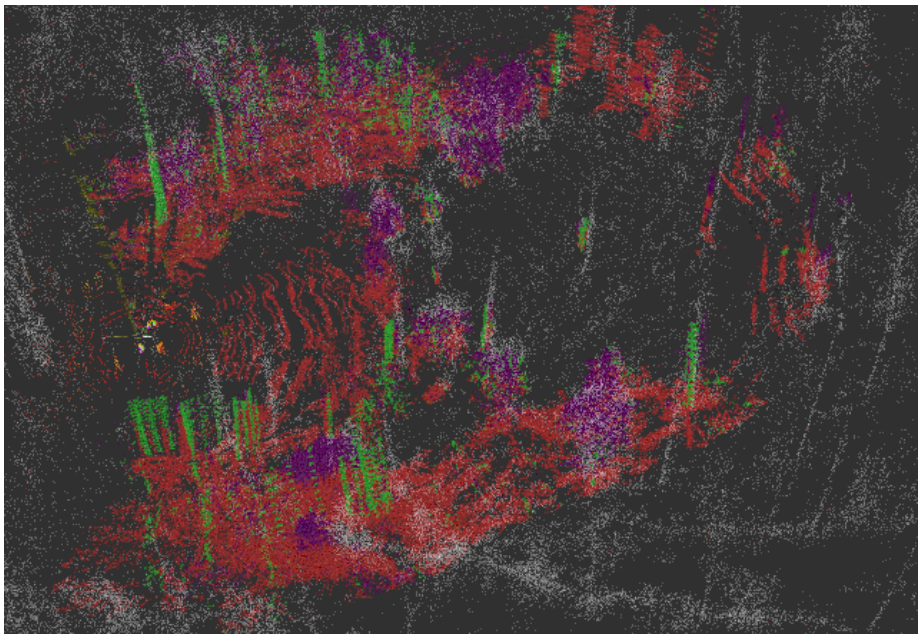


Figure 5.11: Semantic mapping results on the Oporto test site. Fuel is red, trunks are green, canopy is purple, and background is black. White points are unlabeled.

coarse classes, fuel, canopy, and background, which included bare-earth and other non-flammable material. Trunks were not included because they could almost never be observed in the top-down view from the orthomosaic. This manual process took approximately eight hours to complete and is visualized on the left side of Figure 5.12.

In this experiment, we used the SegNext model trained on the manually-labeled dataset from the flight that was not used for mapping. We used the localization LIO-SAM [93] and provided by Franciso Yandun to estimate the pose. In this case, we used UFOMap [33] instead of octomap to aggregate the observations, because of the improved performance. This change was implemented by Duda Andrada and she conducted the experiments reported here. The results of this are shown in Figure 5.12. The overall structure of the scene matches well and small local features are correctly identified. One major source of error is seen in the misregistration of data from adjacent drone flights. This is seen in the vertical offsets that are especially visible on the left side of the map. This may be due to latency in the processing or also because of using the un-refined pose as described previously. The quality of this

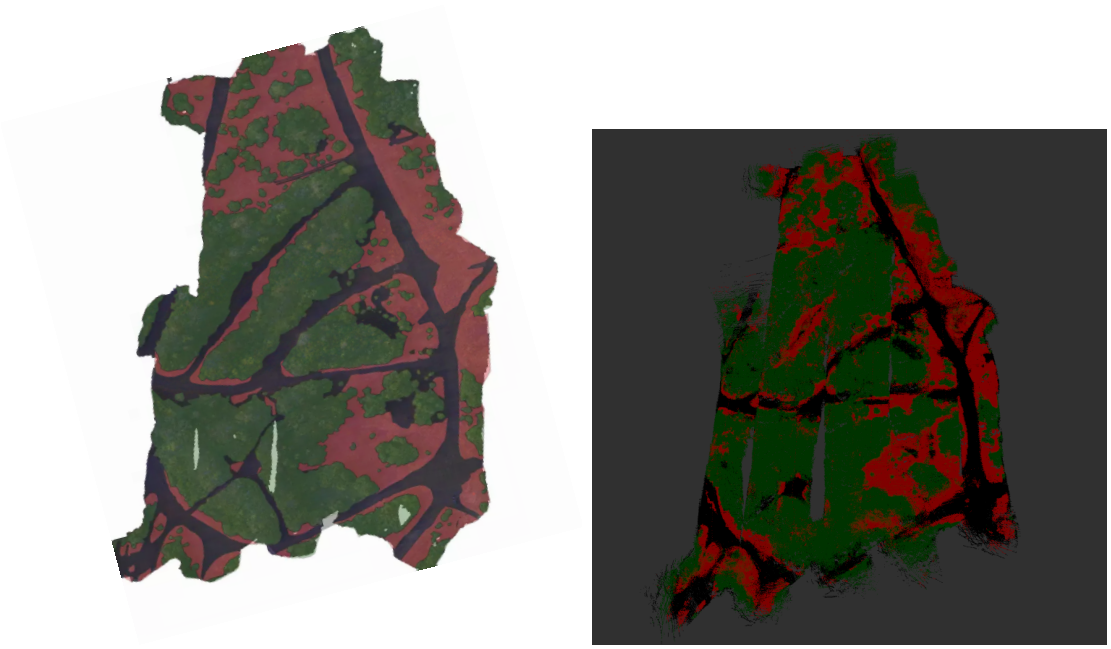


Figure 5.12: Manually-labeled result on the left, results from UFO-Map on the right provided by M. Duda Andrada, using the SegNext [42] model trained on Gascola Data. Taken from [7].

result suggests that with additional work to properly register the scans, this system could be a powerful tool for vegetation mapping.

5.5 Informative Path Planning for Commodity Drones

The goal of this experiment is to simulate a land cover mapping mission for a region that is too large for the drone to exhaustively survey. Remote sensing data is available beforehand and is used to both inform the mission and generate predictions on the regions that are not observed by the drone. In this experiment, we use NAIP data [103] and manual land cover classification annotations from the Chesapeake Bay [20].

In these experiments, we use a simple prediction system to predict the class of unobserved pixels. It is simply a k nearest-neighbor (k=7) classifier that operates on the same PCA-compressed MOSAIKS features that are used for planning. This approach is well-suited to the extremely low number of training samples used in this

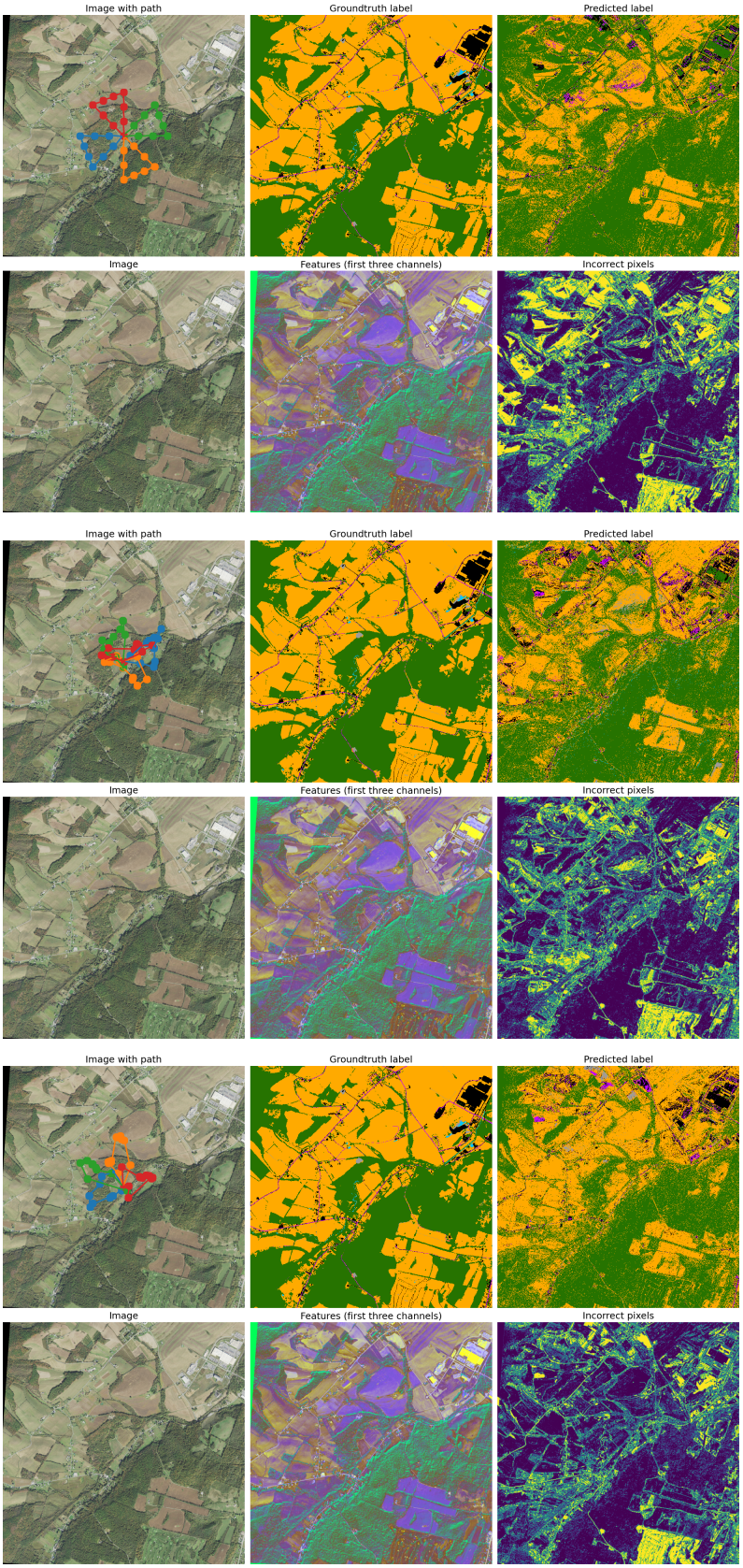


Figure 5.13: Visualizations of the coverage, RAPTORS, and RAPTORS_rare planners from top to bottom. Each color represents an individual flight and they were executed in the following order: blue, orange, green, then red.

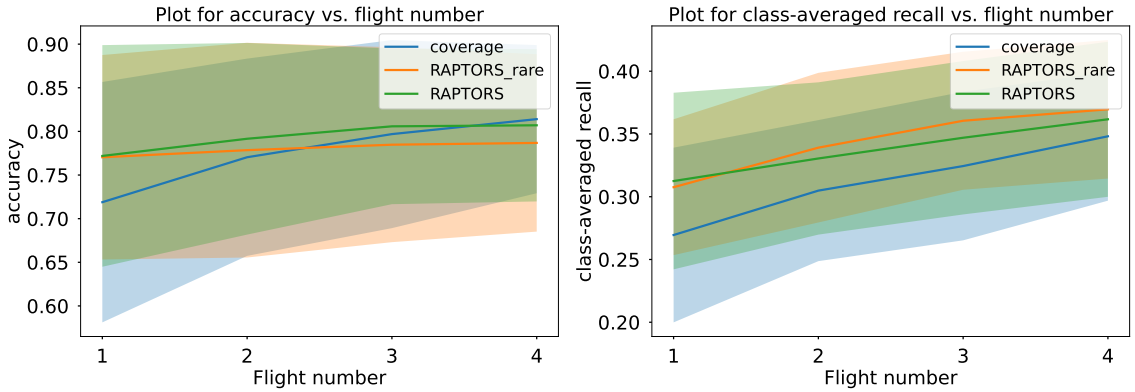


Figure 5.14: Quantitative statistics computed after each flight and averaged across 10 missions. The shaded regions represent the first standard deviation. The left result shows the accuracy and the right shows the class-averaged recall. In both cases higher is better. The proposed methods are only better in terms of accuracy on early flights and the performance converges with the coverage baseline after all flights have been completed. However, both RAPTORS variants achieve a better recall of rare classes after any number of flights. The RAPTORS_rare planner does better class-averaged recall but worse on total accuracy due to prioritizing rare classes. Note that the variance of all methods is high.

setting and the standardized and uncorrelated nature of our feature space.

Before any missions have been executed, the agent can only observe the label of the pixel it is at. Then it plans a mission and executes it, observing the labels at the chosen sampling locations. These samples are used to train a prediction model which is used for evaluation and can be used to inform the plan for the next mission.

The experiments were conducted on 36 random crops drawn from the region in question. Each of the approaches is run on all of the 36 regions individually. Each crop was 6000x6000 pixels, representing a square with 3.6km sides. The agent starts and ends each mission in the center and the path length was set to 3000 pixels (1.8km). This meant that the agent could go halfway to one side of the environment and return to the start within the budget but a significant portion of the domain was unreachable. Each domain was explored using four missions where 10 samples could be collected during each one. Each sample meant the agent could observe the class of a 31x31 pixel (18 meters) square region. After each mission, the class of all pixels is predicted using the KNN classifier, and the error metrics are computed.

The quality of the predictions is evaluated on two metrics: accuracy and averaged recall. The first is the fraction of pixels in the map that were assigned the correct class label by the prediction system. The second represents the average of the per-class recalls. This metric is chosen so that rare classes are treated equally in the evaluation procedure since this is critically important when we explicitly want to find rare classes. We also report the time taken to generate the plan. Note that this does not include the time taken to generate the class predictions, since the planner is agnostic to the choice of prediction algorithm.

The proposed planner is compared to a hand-designed coverage approach. The path consists of a triangular pattern that begins and ends at the central location. This triangle is sized appropriately to exhaust the available path budget. The goal of this path is to cover a diverse set of locations after the four mapping missions have been completed. This planner is visualized alongside the RAPTORS methods in Figure 5.13. As seen there, the RAPTORS methods prioritize collecting a diverse set of samples that span the majority types of land cover.

Quantitative evaluation results are presented Figure 5.14. For all approaches, the performance improves as more samples are collected. The RAPTORS approaches have slightly better total accuracy than the coverage planner in initial flights, but the performance converges over time. This suggests that the quality of the predictions is saturating, especially for the well-represented classes. This is unsurprising because KNN is a simple prediction model. When these approaches are evaluated on class-averaged recall, which weights rare classes equally, the RAPTORS_rare method does better for any number of flights. This suggests that it is able to find a diverse set of initial samples and then leverage the predictions after each flight to prioritize sampling regions predicted to be rare classes. However, across all the experiments the scale of the variance is much higher than the difference between approaches. This variance partially captures the difference in difficulty between the different random sites that the approach was evaluated on. This highlights the challenge of evaluating informative path planning in a rigorous manner. The planning time averaged across the RAPTORS variants was 252.2 seconds per path. These experiments were executed on high-end cloud infrastructure, but this shows that the approach has the potential to be deployed in a realistic field scenario in the future.

Chapter 6

Conclusions

The goal of this work is to take initial steps toward integrated intelligent forest management. The experiments we conducted span a variety of topics, but some common themes emerged. The first is that structure from motion is a powerful tool for drone-based mapping and should likely be explored before resorting to more complex multi-sensor SLAM solutions. The second general trend is that often the most effective way to generate deep learning predictions is by annotating a small amount of high-quality and relevant data, rather than relying on higher volumes of lower-quality annotations. This suggests that specializing individual models to a given scene is more promising than trying to have one static model that covers the full variability of a task. Finally, these experiments show significantly more compelling results when using drone data as opposed to remote sensing data. As remote sensing technology improves, the role of drones may deminish, but this scenario appears unlikely in the near future.

6.1 Key Takeaways

We find that the structure from motion parameters suggested by Young et. al. [114] work well for reconstructing a number of diverse datasets. The performance is the best when the drone conducts a lawnmower survey above the canopy but manual non-overlapping flight can produce acceptable results. It is more challenging to generate reconstructions from images taken under the canopy, but further research

may be able to address this issue.

We compare the performance of tree detection from both drone and NAIP data. The detections from drone data are still significantly better than those from NAIP data, even after resampling to the same resolution. However, using drone data downsampled to the same resolution as NAIP yields almost as good performance as the original orthomosaic resolution.

This work demonstrates an approach to map the location of different types of vegetation using a drone equipped with a camera and LiDAR. We show that recent transformer-based semantic segmentation models are able to achieve moderate performance when trained on very few images from a target region. We show that the local structure of these maps is accurate, but the global structure is highly dependent on the quality of the SLAM predictions.

We show that unsupervised feature extraction is a powerful tool for land-cover classification from remote sensing data. Specifically, using the approach described in MOSAIKS and compressing these features with PCA yields a compact and informative representation. These features can be used to plan informative drone flights by choosing samples that minimize Gaussian Process uncertainty about the whole region. This approach results in a very small improvement in accuracy for the first flights but this converges after all four flights have been conducted. The proposed approach is better than a coverage planner at identifying rare classes, no matter how many flights have been executed. However, it's important to note that the variance in prediction quality is high for both metrics and further study is required to draw rigorous conclusions.

This can be summarized briefly by a set of techniques that we found effective:

- Infering 3D geometry with SfM on over-canopy drone data
- Infering 3D geometry with a well-tuned SLAM system on under-canopy drone data
- Detecting trees in drone-derived ortho-mosaics using deep learning
- Training semantic segmentation models on low numbers of images
- Performing semantic mapping with image-based semantic segmentation and
- Extracting semantically-meaningful unsupervised features using MOSAIKS [83]

and PCA

The following set of techniques yielded unsatisfactory or inconclusive results:

- SfM in under-canopy environments
- Training tree detection models on aerial data from NAIP using either manual annotations or drone predictions
- Informative path planning for dramatically-better land cover mapping sample selection

6.2 Contributions

This work makes several contributions:

- We collect drone datasets in a variety of forest settings and implement software infastructure to process this data.
- We conduct experiments evaluating the performance of tree detection on drone and remote sensing data.
- We present a system for mapping different types of vegetation using multi-senors SLAM for geometric information and vision-based semantic segmentation to differentiate vegetation classes. This has applications to forest fire mitigation and we believe this is the first system of its kind to address this problem.
- We propose a novel long-horizon informative path planner that is applicable for planning sparse surveys with commodity drones. We demonstrate that this method is helpful for choosing samples for a land-cover classification task when it is important to identify rare classes.

6.3 Future Work

From our tree detection experiments, we conclude that applying existing tree detectors to NAIP data does not result in useful predictions but using downsampled drone orthomosaics does. This suggests that exploring other satellite data products could

be promising. Alternatively, approaches to make the properties of the NAIP data more similar to the orthomosaics could be used. For example, by manually adjusting the saturation or sharpness, or using a learning-based approach.

In this work, we have demonstrated an online semantic mapping system that uses a custom multi-sensor payload. A logical extension to this method would be using only GPS-tagged images from commodity drones as input. Fortunately, SfM operating on these images can be used to replace the geometric reasoning provided by SLAM. Aggregating multi-view semantic information with meshes has been shown to be an effective strategy for 3D semantic mapping. As an additional improvement, geometric data from the meshes could be used to augment the visual data for semantic segmentation. Rendering techniques could be used to compute which point on the mesh corresponds to each pixel in the image. Geometric information from the mesh, such as height, could then be added as an additional input to the semantic segmentation network.

One way to integrate all the themes in this work would be to conduct real-world informative path planning experiments. This would involve planning a path with RAPTORS, classifying the drone data with the demonstrated semantic mapping or proposed semantic meshes approach, and then generating predictions from unsupervised features using the drone plots as training samples. The quality of these predictions could be assessed either by using an existing dataset as ground truth or by flying an exhaustive drone survey and using this as a pseudo-ground truth.

A central theme of this work is conducting experiments where limited ground truth data exists. This makes thorough analysis challenging and limits participation in forestry robotics. Future research should focus on collecting annotated datasets that are informed by best practices from both the robotics and ecology communities. This will ensure that a diverse set of technical experiments are possible and that the questions they address provide ecologically relevant answers.

Bibliography

[1] Sameer Agarwal, Noah Snavely, Ian Simon, Steven M. Seitz, and Richard Szeliski. Building Rome in a day. *Proceedings of the IEEE International Conference on Computer Vision*, (Iccv):72–79, 2009. doi: 10.1109/ICCV.2009.5459148. 2.3.1

[2] Agisoft. Agisoft metashape. URL https://www.agisoft.com/. (document), 2.3.1, 2.1, 4.2.1

[3] United States Department of Agriculture. Hazardous fuels treatments slow fire advance on HK Complex. *Online*, 2019. URL https://www.fs.usda.gov/sites/default/files/2020-02/20190905_hk_complex_fuels_treatment_success_story_final.pdf. 2.1.2

[4] Hal E Anderson. *Aids to determining fuel models for estimating fire behavior*, volume 122. US Department of Agriculture, Forest Service, Intermountain Forest and Range . . . , 1981. (document), 4.4.1, 4.3, 5.2

[5] M E Andrada, J F Ferreira, D Portugal, and M Couceiro. Testing Different CNN Architectures for Semantic Segmentation for Landscaping with Forestry Robotics. *In: IROS 2020 Workshop on Perception, Planning and Mobility in Forestry Robotics (WPPMFR 2020)*, 2020. 4.1.3, **??**, 4.4

[6] M Eduarda Andrada, Joao F Ferreira, and Micael S Couceiro. Integration of an Artificial Perception System for Identification of Live Flammable Material in Forestry Robotics. 2022. 3.2

[7] Maria Eduarda Andrada, David Russell, Tito Arevalo-Ramirez, Winnie Kuang, George Kantor, and Francisco Yandun. Mapping of Potential Fuel Regions Using Uncrewed Aerial Vehicles for Wildfire Prevention. *Forests*, 14(8), 2023. ISSN 1999-4907. doi: 10.3390/f14081601. URL https://www.mdpi.com/1999-4907/14/8/1601. (document), 3.2, 5.2, 5.12

[8] Martin Baatz and Arno Schäpe. Multiresolution segmentation : an optimization approach for high quality multi-scale image segmentation. 2000. 3.1

[9] Grayson Badgley, Jeremy Freeman, Joseph J. Hamman, Barbara Haya, Anna T. Trugman, William R.L. Anderegg, and Danny Cullenward. Systematic overcrediting in California's forest carbon offsets program. *Global Change Biology*,

28(4):1433–1445, 2 2022. ISSN 13652486. doi: 10.1111/gcb.15943. 2.1.1

[10] James G. C. Ball, Sebastian H. M. Hickman, Tobias D. Jackson, Xian Jing Koay, James Hirst, William Jay, Mélaine Aubry-Kientz, Grégoire Vincent, and David A. Coomes. Accurate tropical forest individual tree crown delineation from RGB imagery using Mask R-CNN. *bioRxiv*, pages 1–18, 2022. doi: 10.1002/rse2.332. URL https://www.biorxiv.org/content/10.1101/2022.07.10.499480v1%0Ahttps://www.biorxiv.org/content/10.1101/2022.07.10.499480v1.abstract. 3.1

[11] Martin Beland, Geoffrey Parker, Ben Sparrow, David Harding, Laura Chasmer, Stuart Phinn, Alexander Antonarakis, and Alan Strahler. On promoting the use of lidar systems in forest ecosystem research. *Forest Ecology and Management*, 450, 08 2019. doi: 10.1016/j.foreco.2019.117484. 2.2.3

[12] Jonathan Binney, Andreas Krause, and Gaurav S. Sukhatme. Optimizing waypoints for monitoring spatiotemporal phenomena. *International Journal of Robotics Research*, 32(8):873–888, 2013. ISSN 02783649. doi: 10.1177/0278364913488427. 3.3

[13] Christopher Blaufelder, Cindy Levy, Peter Mannion, and Dickon Pinner. A Blueprint for Scaling Voluntary Carbon Markets to Meet the Climate Challenge. *McKinsey & Company*, (January):7, 2021. 2.1.1

[14] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A Survey of Monte Carlo Tree Search Methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1):1–43, 2012. ISSN 1943-068X. 3.3.1, 3.3.2

[15] Alberto Candela, Suhit Kodgule, Kevin Edelson, Srinivasan Vijayarangan, David R. Thompson, Eldar Noe Dobrea, and David Wettergreen. Planetary Rover Exploration Combining Remote and in Situ Measurements for Active Spectroscopic Mapping. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 5986–5993, 2020. ISSN 10504729. doi: 10.1109/ICRA40945.2020.9196973. 3.3.2, 3.4, 4.5, 4.5.1

[16] Alberto Candela, Kevin Edelson, Michelle M. Gierach, David R. Thompson, Gail Woodward, and David Wettergreen. Using Remote Sensing and in situ Measurements for Efficient Mapping and Optimal Sampling of Coral Reefs. *Frontiers in Marine Science*, 8(September):1–17, 2021. ISSN 22967745. doi: 10.3389/fmars.2021.689489. 4.5.1

[17] National Interagency Fire Center. National fire news, 2022. URL https://www.nifc.gov/fire-information/nfn. https://www.nifc.gov/

fire-information/nfn(Accessed: March 2022). 2.1.2

[18] Steven W. Chen, Guilherme V. Nardari, Elijah S. Lee, Chao Qu, Xu Liu, Roseli Ap Francelin Romero, and Vijay Kumar. SLOAM: Semantic Lidar Odometry and Mapping for Forest Inventory. *IEEE Robotics and Automation Letters*, 5 (2):612–619, 2020. ISSN 23773766. doi: 10.1109/LRA.2019.2963823. 2.3.1, 4.4, 5.4.2

[19] Yangyang Chen, Dongping Ming, Lu Zhao, Beiru Lv, Keqi Zhou, and Yuanzhao Qing. Review on high spatial resolution remote sensing image segmentation evaluation. *Photogrammetric Engineering and Remote Sensing*, 84(10):629–646, 2018. ISSN 00991112. doi: 10.14358/PERS.84.10.629. 3.1

[20] Peter Claggett, Labeeb Ahmed, Ernie Buford, Jacob Czawlytko, Sean Macfaden, Patrick Mccabe, Sarah Mcdonald, Jarlath O Neill, Anna Royar, Kelly Schulze, and Katie Walker. Chesapeake Bay Program ' s One -meter Resolution Land Use / Land Cover Data : Overview and Production. 2014. ??, 5.5

[21] State of California Public Utilities Commision. Global Strategies for Utility Wildfire Mitigation Utility Wildfire Mitigation Strategy and Roadmap for the Wildfire Safety Division. *Online*, pages 1–28. URL https://www.cbsnews.com/news/amazon-wildfires-brazil-spurns-20-million-aid-offer-from-g-7-nations-today-2019 . 2.1.2

[22] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 4.4.1

[23] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem:3213–3223, 2016. ISSN 10636919. doi: 10.1109/CVPR.2016.350. 5.4.1

[24] Micael S. Couceiro, David Portugal, Joao F. Ferreira, and Rui P. Rocha. SEMFIRE: Towards a new generation of forestry maintenance multi-robot systems. *Proceedings of the 2019 IEEE/SICE International Symposium on System Integration, SII 2019*, pages 270–276, 2019. doi: 10.1109/SII.2019.8700403. 3.2

[25] Micael S Couceiro, David Portugal, João F Ferreira, and Rui P Rocha. Semfire: Towards a new generation of forestry maintenance multi-robot systems. In *2019 IEEE/SICE International Symposium on System Integration (SII)*, pages 270–276. IEEE, 2019. 2.1.2, 3.4

[26] Matthew B Creasy, Wade T Tinkham, Chad M Hoffman, and Jody C Vogeler. Potential for individual tree monitoring in ponderosa pine dominated forests using unmanned aerial system structure from motion point clouds. *Canadian Journal of Forest Research*, 51(8):1093–1105, 2021. doi: 10.1139/cjfr-2020-0433. URL https://doi.org/10.1139/cjfr-2020-0433. 2.3.1

[27] Richard Cristan and Arjun Rijal. Drones in Forest Management. *Alabama Cooperative Extension System*, pages 23–24. 2.2.2

[28] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1, 2005. doi: 10.1109/CVPR.2005.177. 2.3.2

[29] Daniel Davila, Joseph Vanpelt, Alexander Lynch, Adam Romlein, Peter Webley, and Matthew S Brown. ADAPT: An Open-Source sUAS Payload for Real-Time Disaster Prediction and Response with AI. Technical report, 2022. URL https://kitware.github.io/adapt/. 4.4.1

[30] Matthew Dawkins, Linus Sherrill, Keith Fieldhouse, Anthony Hoogs, Benjamin Richards, David Zhang, Lakshman Prasad, Kresimir Williams, Nathan Lauffenburger, and Gaoang Wang. An open-source platform for underwater image & video analytics. *Proceedings - 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017*, pages 898–906, 2017. doi: 10.1109/WACV.2017.105. 4.4.1

[31] Frank Dellaert and Michael Kaess. Factor Graphs for Robot Perception. *Foundations and Trends in Robotics*, 6(1-2):1–139, 2017. ISSN 1935-8253. doi: 10.1561/2300000043. 2.3.1, 5.4.2

[32] Drone Deploy. Drone deploy. URL https://help.dronedeploy.com/hc/en-us. 2.2.2

[33] Daniel Duberg and Patric Jensfelt. UFOMap: An Efficient Probabilistic 3D Mapping Framework That Embraces the Unknown. *IEEE Robotics and Automation Letters*, 5(4):6411–6418, 2020. ISSN 23773766. doi: 10.1109/LRA.2020.3013861. 5.4.2

[34] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006. 2.3.1

[35] Kevin Edelson. Ergodic Trajectory Optimization for Information Gathering. 2020. 3.3.2

[36] Isabel M.Rayas Fernandez, Christopher E. Denniston, David A. Caron, and Gaurav S. Sukhatme. Informative Path Planning to Estimate Quantiles for Environmental Analysis. *IEEE Robotics and Automation Letters*, 7(4):10280–10287, 2022. ISSN 23773766. doi: 10.1109/LRA.2022.3191936. 4.5.1

[37] Office of Wildland Fire. Fuels Management. *Online*, pages 313–326, 2021. URL https://www.doi.gov/wildlandfire/fuels#:~:text=Thinningforestedareaswithchainsaws,withinvasiveplantsusingherbicides. 2.1.2, 4.4

[38] Benjamin T. Fraser and Russell G. Congalton. A comparison of methods for determining forest composition from high-spatial-resolution remotely sensed imagery. *Forests*, 12(9), 2021. ISSN 19994907. doi: 10.3390/f12091290. 3.1

[39] Michael Gillenwater and Stephen Seres. The Clean Development Mechanism: a review of the first international offset programme. *Greenhouse Gas Measurement and Management*, 1(3-4):179–203, 2011. ISSN 2043-0779. doi: 10.1080/20430779.2011.647014. 2.1.1

[40] Fillipe Goulart, 2023. 4.5.2

[41] Bronson W. Griscom, Justin Adams, Peter W. Ellis, Richard A. Houghton, Guy Lomax, Daniela A. Miteva, William H. Schlesinger, David Shoch, Juha V. Siikamäki, Pete Smith, Peter Woodbury, Chris Zganjar, Allen Blackman, João Campari, Richard T. Conant, Christopher Delgado, Patricia Elias, Trisha Gopalakrishna, Marisa R. Hamsik, Mario Herrero, Joseph Kiesecker, Emily Landis, Lars Laestadius, Sara M. Leavitt, Susan Minnemeyer, Stephen Polasky, Peter Potapov, Francis E. Putz, Jonathan Sanderman, Marcel Silvius, Eva Wollenberg, and Joseph Fargione. Natural climate solutions. *Proceedings of the National Academy of Sciences of the United States of America*, 114(44):11645–11650, 2017. ISSN 10916490. doi: 10.1073/pnas.1710465114. 2.1.1

[42] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zhengning Liu, Ming-Ming Cheng, and Shi-Min Hu. SegNeXt: Rethinking Convolutional Attention Design for Semantic Segmentation. (NeurIPS):1–15, 2022. URL http://arxiv.org/abs/2209.08575. (document), 2.3.2, 5.2, 5.12

[43] Cynthia Hall. What is Synthetic Aperture Radar? — Earthdata. *NASA Earthdata*, pages 1–12, 2020. URL https://www.earthdata.nasa.gov/learn/backgrounders/what-is-sar. 2.2.3

[44] M. C. Hansen, P. V. Potapov, R. Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland, A. Kommareddy, A. Egorov, L. Chini, C. O. Justice, and J. R.G. Townshend. High-resolution global maps of 21st-century forest cover change. *Science*, 342(6160):850–853, 2013. ISSN 10959203. doi: 10.1126/science.1244693. 2.2.3

[45] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. 3 2017. URL http://arxiv.org/abs/1703.06870. 2.3.2, 3.1

[46] John S Hogland, Nathaniel M Anderson, Woodam Chung, and Lucas Wells. Estimating forest characteristics using NAIP imagery and ArcOb-

jects. *Proceedings of the 2014 ESRI Users Conference; July 14-18, 2014, San Diego, CA. Redlands, CA: Environmental Systems Research Institute. Online: http://proceedings.esri.com/library/userconf/proc14/papers/155_181.pdf*, pages 1–22, 2014. 2.2

[47] Geoffrey A. Hollinger and Gaurav S. Sukhatme. Sampling-based robotic information gathering algorithms. *International Journal of Robotics Research*, 33 (9):1271–1287, 2014. ISSN 17413176. doi: 10.1177/0278364914533443. 3.3.1

[48] Armin Hornung, Kai M. Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 2013. doi: 10.1007/s10514-012-9321-0. URL https://octomap.github.io. Software available at https://octomap.github.io. 3.2, 4.4.2

[49] April Hulet, Bruce A. Roundy, Steven L. Petersen, Stephen C. Bunting, Ryan R. Jensen, and Darrell B. Roundy. Utilizing National Agriculture Imagery Program Data to Estimate Tree Cover and Biomass of Piñon and Juniper Woodlands. *Rangeland Ecology and Management*, 67(5):563–572, 2014. ISSN 15507424. doi: 10.2111/REM-D-13-00044.1. 3.1

[50] IPCC. Climate Change and Land: an IPCC special report. *Climate Change and Land: an IPCC Special Report on climate change, desertification, land degradation, sustainable land management, food security, and greenhouse gas fluxes in terrestrial ecosystems*, pages 1–864, 2019. URL https://www.ipcc.ch/srccl/. 1

[51] Neal Jean, Sherrie Wang, Anshul Samar, George Azzari, David Lobell, and Stefano Ermon. Tile2Vec: Unsupervised representation learning for spatially distributed data. *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, pages 3967–3974, 2019. ISSN 2159-5399. doi: 10.1609/aaai.v33i01.33013967. 4.5

[52] Michael J Jenkins, Justin B Runyon, Christopher J Fettig, Wesley G Page, and Barbara J Bentz. and Fuels. 60(June):489–501, 2014. 2.1.2

[53] Ian T. Jollife and Jorge Cadima. Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065), 2016. ISSN 1364503X. doi: 10.1098/rsta.2015.0202. 4.5

[54] Robert S. Kaplan, Karthik Ramanna, and Marc Roston. Accounting for Carbon Offsets – Establishing the Foundation for Carbon-Trading Markets. *SSRN Electronic Journal*, 2023. doi: 10.2139/ssrn.4383510. 2.1.1

[55] Michael Keller, David S. Schimel, William W. Hargrove, and Forrest M. Hoffman. A continental strategy for the National Ecological Observatory Network. *Frontiers in Ecology and the Environment*, 6(5):282–284, 2008. ISSN 15409295. doi: 10.1890/1540-9295(2008)6[282:ACSFTN]2.0.CO;2. 3.1, 4.3

[56] Pieter Kempeneers, Fernando Sedano, Lucia Seebach, Peter Strobl, and Jesús San-Miguel-Ayanz. Data fusion of different spatial resolution remote sensing images applied to forest-type mapping. *IEEE Transactions on Geoscience and Remote Sensing*, 49(12 PART 2):4977–4986, 2011. ISSN 01962892. doi: 10.1109/TGRS.2011.2158548. 2.2.3

[57] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 4.3

[58] Suhit Kodgule, Alberto Candela, and David Wettergreen. Non-myopic Planetary Exploration Combining in Situ and Remote Measurements. *IEEE International Conference on Intelligent Robots and Systems*, pages 536–543, 2019. ISSN 21530866. doi: 10.1109/IROS40897.2019.8967769. 3.3.2

[59] Ioannis Kostavelis and Antonios Gasteratos. Semantic mapping for mobile robotics tasks: A survey. *Robotics and Autonomous Systems*, 66:86–103, 2015. ISSN 09218890. doi: 10.1016/j.robot.2014.12.006. URL http://dx.doi.org/10.1016/j.robot.2014.12.006. 3.2

[60] Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9:235–284, 2008. ISSN 15324435. 4.5.1

[61] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F Pereira, C J Burges, L Bottou, and K Q Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf. 2.3.2

[62] Tiina Kurvits, Alexandra Popescu, Alison Paulson, Andrew Sullivan, David Ganz, Chantelle Burton, Douglas Kelley, Paulo Fernandes, Lea Wittenberg, Elaine Baker, Patrícia S. Silva, Camilla Mathison, Dolors Armenteras, and Bibiana Bilbao. *Spreading like wildfire: the rising threat of extraordinary landscape fires*. 03 2022. 2.1.2

[63] Planet Labs. Planet labs. URL https://www.planet.com/. 4.1.4

[64] LANDFIRE. Value of Georeferenced Vegetation Plots to LANDFIRE. *Project report*, (January), 2018. 4.5

[65] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521 (7553):436–444, 2015. ISSN 14764687. doi: 10.1038/nature14539. 2.3.2

[66] Tsung Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327, 2020. ISSN 19393539. doi: 10.1109/TPAMI.2018.2858826. 2.3.2, 3.1

[67] George Mathew and Igor Mezic. Metrics for ergodicity and design of ergodic dynamics for multi-agent systems. *Physica D: Nonlinear Phenomena*, 240, 02 2011. doi: 10.1016/j.physd.2010.10.010. 3.3.1

[68] Landsat Missions. Landsat satellite missions. URL https://www.usgs.gov/landsat-missions/landsat-satellite-missions. 2.2.3

[69] Brady Moon, Satrajit Chatterjee, and Sebastian Scherer. TIGRIS: An Informed Sampling-based Algorithm for Informative Path Planning. *IEEE International Conference on Intelligent Robots and Systems*, 2022-Octob:5760–5766, 2022. ISSN 21530866. doi: 10.1109/IROS47612.2022.9981992. 3.3.1

[70] Rui Jose Silva Oliveira Nunes. *Procedural Generation of Synthetic Forest Environments to Train Machine Learning Algorithms*. PhD thesis, Universidade de Coimbra, 2021. 4.1.3, ??

[71] S.N. Oswalt, W.B. Smith, P.D. Miles, and S.A. Pugh. Update of the 2010 RPA Assessment Forest Resources of the United States , 2012 :. (October):228, 2014. 2.2

[72] Panos M Pardalos and Thelma D Mavridou. Simulated annealingSimulated Annealing. In Christodoulos A Floudas and Panos M Pardalos, editors, *Encyclopedia of Optimization*, pages 3591–3593. Springer US, Boston, MA, 2009. ISBN 978-0-387-74759-0. doi: 10.1007/978-0-387-74759-0{_}617. URL https://doi.org/10.1007/978-0-387-74759-0_617. 4.5.2

[73] Lorena Parra. Remote Sensing and GIS in Environmental Monitoring. *Applied Sciences (Switzerland)*, 12(16), 2022. ISSN 20763417. doi: 10.3390/app12168045. 2.2.3

[74] Sorin C. Popescu and Randolph H. Wynne. Seeing the trees in the forest: Using lidar and multispectral data fusion with local filtering and variable window size for estimating tree height. *Photogrammetric Engineering and Remote Sensing*, 70(5):589–604, 2004. ISSN 00991112. doi: 10.14358/PERS.70.5.589. 3.1

[75] Marija Popović, Teresa Vidal-Calleja, Gregory Hitz, Jen Jen Chung, Inkyu Sa, Roland Siegwart, and Juan Nieto. An informative path planning framework for UAV-based terrain monitoring. *Autonomous Robots*, 44(6):889–911, 2020. ISSN 15737527. doi: 10.1007/s10514-020-09903-2. 3.3.2

[76] QGIS Development Team. *QGIS Geographic Information System*. QGIS Association, 2023. URL https://www.qgis.org. 4.3, 5.4.2

[77] QGroundControl. Qgroundcontrol. URL http://qgroundcontrol.com/. 2.2.2

[78] Ananya Rao, Ian Abraham, Guillaume Sartoretti, and Howie Choset. Sparse Sensing in Ergodic Optimization. pages 1–14. 3.3.1

[79] Carl Edward Rasmussen. Gaussian Processes in Machine Learning. In Olivier Bousquet, Ulrike von Luxburg, and Gunnar Rätsch, editors, *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures*, pages 63–71. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. ISBN 978-3-540-28650-9. doi: 10.1007/978-3-540-28650-9{_}4. URL https://doi.org/10.1007/978-3-540-28650-9_4. 3.3.2, 4.5.1

[80] Gyri Reiersen, David Dao, Björn Lütjens, Konstantin Klemmer, Kenza Amara, Attila Steinegger, Ce Zhang, and Xiaoxiang Zhu. ReforesTree: A Dataset for Estimating Tropical Forest Carbon Stock with Deep Learning and Aerial Imagery. 2022. URL http://arxiv.org/abs/2201.11192. 2.1.1

[81] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017. ISSN 01628828. doi: 10.1109/TPAMI.2016.2577031. 2.3.2

[82] Caleb Robinson, Le Hou, Kolya Malkin, Rachel Soobitsky, Jacob Czawlytko, Bistra DIlkina, and Nebojsa Jojic. Large scale high-resolution land cover mapping with multi-resolution data. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:12718–12727, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.01301. ??

[83] Esther Rolf, Jonathan Proctor, Tamma Carleton, Ian Bolliger, Vaishaal Shankar, Miyabi Ishihara, Benjamin Recht, and Solomon Hsiang. A generalizable and accessible approach to machine learning with global satellite imagery. *Nature Communications*, 12(1):1–11, 2021. ISSN 20411723. doi: 10.1038/s41467-021-24638-z. URL http://dx.doi.org/10.1038/s41467-021-24638-z. 4.5, 6.1

[84] Adriana Romero, Carlo Gatta, and Gustau Camps-Valls. Unsupervised deep feature extraction for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 54(3):1349–1362, 2016. ISSN 01962892. doi: 10.1109/TGRS.2015.2478379. 4.5

[85] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. URL http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a. (available on arXiv:1505.04597 [cs.CV]). 2.3.2

[86] Antoni Rosinol, Marcus Abate, Yun Chang, and Luca Carlone. Kimera: An

Open-Source Library for Real-Time Metric-Semantic Localization and Mapping. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 1689–1696, 2020. ISSN 10504729. doi: 10.1109/ICRA40945.2020.9196885. 3.2

[87] Julius Rückin, Liren Jin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Informative Path Planning for Active Learning in Aerial Semantic Mapping. 2022. URL http://arxiv.org/abs/2203.01652. 3.3.2, 3.4

[88] David Russell, Tito Arevalo, Chinmay Garg, Winnie Kuang, Francisco Yandun, David Wettergreen, and George Kantor. Unmanned Aerial Vehicle Mapping with Semantic and Traversability Metrics for Forest Fire Mitigation. Technical report. 3.2, 4.2.2, 5.4.2

[89] J. San-Miguel-Ayanz, T. Durrant, R. Boca, P. Maianti, G. Libertá, T. Artés-Vivancos, D. Oom, A. Branco, D. de Rigo, D. Ferrari, H. Pfeiffer, R. Grecchi, D. Nuijten, and M. Onida. Advance effis report on forest fires in europe, middle east and north africa 2020, 2021. URL https://effis-gwis-cms.s3-eu-west-1.amazonaws.com/effis/reports-and-publications/effis-related-publications/Advance_EFFIS_Report+on+Forest+Fires+in+Europe_2020_210401xv_finalv2.pdf. 2.1.2

[90] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2.3.1

[91] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 2.3.1

[92] Washington Forest Service. FSH 2409.12 - TIMBER CRUISING HANDBOOK. (61). 2.2.1

[93] Tixiao Shan, Brendan Englot, Drew Meyers, Wei Wang, Carlo Ratti, and Daniela Rus. LIO-SAM: Tightly-coupled lidar inertial odometry via smoothing and mapping. *IEEE International Conference on Intelligent Robots and Systems*, pages 5135–5142, 2020. ISSN 21530866. doi: 10.1109/IROS45743.2020.9341176. 2.3.1, 4.2.2, 5.2, 5.4.2

[94] Weizhao Shao, Srinivasan Vijayarangan, Cong Li, and George Kantor. Stereo Visual Inertial LiDAR Simultaneous Localization and Mapping. *IEEE International Conference on Intelligent Robots and Systems*, pages 370–377, 2019. ISSN 21530866. doi: 10.1109/IROS40897.2019.8968012. 4.2.2

[95] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis*

and Machine Intelligence, 39(4):640–651, 2017. ISSN 01628828. doi: 10.1109/TPAMI.2016.2572683. 2.3.2

[96] Felix Stache, Jonas Westheider, Federico Magistri, Marija Popovic, and Cyrill Stachniss. Adaptive path planning for UAV-based multi-resolution semantic segmentation. *2021 10th European Conference on Mobile Robots, ECMR 2021 - Proceedings*, 2021. doi: 10.1109/ECMR50962.2021.9568788. 3.3.2

[97] Neal C. Swayze, Wade T. Tinkham, Jody C. Vogeler, and Andrew T. Hudak. Influence of flight parameters on UAS-based monitoring of tree height, diameter, and density. *Remote Sensing of Environment*, 263(May):112540, 2021. ISSN 00344257. doi: 10.1016/j.rse.2021.112540. URL https://doi.org/10.1016/j.rse.2021.112540. 2.3.1

[98] Lina Tang and Guofan Shao. Drone remote sensing for forestry research and practices. *Journal of Forestry Research*, 26(4):791–797, 2015. ISSN 19930607. doi: 10.1007/s11676-015-0088-y. 2.2.2

[99] Lina Tang, Guofan Shao, and Limin Dai. Roles of digital technology in china's sustainable forestry development. *International Journal of Sustainable Development World Ecology*, 16:94–101, 04 2009. doi: 10.1080/13504500902794000. 2.2

[100] Arturo Balderas Torres and Jon C. Lovett. Using basal area to estimate aboveground carbon stocks in forests: La Primavera Biosphere's Reserve, Mexico. *Forestry*, 86(2):267–281, 2013. ISSN 0015752X. doi: 10.1093/forestry/cps084. 2.1.1

[101] Chris Town, Eunice Padley, Brian Kruse, Tom Ward, and Frank Gariglio. Forestry Technical Note No . FOR – 1 Forestry Inventory Methods. *Forestry Inventory Methods*, (July), 2018. 2.2.1

[102] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle adjustment – a modern synthesis. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1883:298–372, 2000. ISSN 16113349. doi: 10.1007/3-540-44480-7{_}21. 2.3.1

[103] U.S. Department of Agriculture. National Agriculture Imagery Program (NAIP) Information Sheet. (March), 2011. URL https://www.fsa.usda.gov/Internet/FSA_File/naip_2010_infosheet.pdf. 3.1, ??, 5.5

[104] US Forest Service Department of Agriculture. FOREST INVENTORY AND ANALYSIS NATIONAL CORE FIELD GUIDE VOLUME I: FIELD DATA COLLECTION PROCEDURES FOR PHASE 2 PLOTS. 2016. 2.2.1

[105] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you

need. *Advances in Neural Information Processing Systems*, 2017-Decem(Nips): 5999–6009, 2017. ISSN 10495258. 2.3.2

[106] Ben G. Weinstein, Sergio Marconi, Mélaine Aubry-Kientz, Gregoire Vincent, Henry Senyondo, and Ethan P. White. DeepForest: A Python package for RGB deep learning tree crown delineation. *Methods in Ecology and Evolution*, 11 (12):1743–1751, 2020. ISSN 2041210X. doi: 10.1111/2041-210X.13472. 3.1, 3.4, 4.3, 4.3

[107] Thales A P West, Jan Börner, Erin O Sills, and Andreas Kontoleon. Overstated carbon emission reductions from voluntary REDD+ projects in the Brazilian Amazon. 117(39):24188–24194, 2020. doi: 10.1073/pnas.2004334117/-/DCSupplemental. 2.1.1

[108] Wildland Fire Resiliency Program. 4 Vegetation Management Plan. pages 1–37, 2021. URL https://www.openspace.org/sites/default/files/4-Vegetation_Management_Plan.pdf. 2.1.2

[109] Wen Xiao, Aleksandra Zaforemska, Magdalena Smigaj, Yunsheng Wang, and Rachel Gaulton. Mean shift segmentation assessment for individual forest tree delineation from airborne lidar data. *Remote Sensing*, 11(11):1–19, 2019. ISSN 20724292. doi: 10.3390/rs11111263. 3.1

[110] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. pages 1–18, 2021. URL http://arxiv.org/abs/2105.15203. 2.3.2, 4.4.1

[111] Michael Xie, Neal Jean, Marshall Burke, David Lobell, and Stefano Ermon. Transfer learning from deep features for remote sensing and poverty mapping. *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, pages 3929–3935, 2016. ISSN 2159-5399. doi: 10.1609/aaai.v30i1.9906. 4.5

[112] Zhang Xuan and Filliat David. Real-time voxel based 3d semantic mapping with a hand held rgb-d camera. https://github.com/floatlazer/semantic_slam, 2018. 4.4.2

[113] Zhang Xuan and Filliat David. Real-time voxel based 3d semantic mapping with a hand held rgb-d camera. https://github.com/floatlazer/semantic_slam, 2018. 3.2, 3.4

[114] Derek J.N. Young, Michael J. Koontz, and Jonah Maria Weeks. Optimizing aerial imagery collection and processing parameters for drone-based individual tree mapping in structurally complex conifer forests. *Methods in Ecology and Evolution*, 13(7):1447–1463, 2022. ISSN 2041210X. doi: 10.1111/2041-210X.13860. 2.3.1, 4.2.1, 5.1, 6.1

[115] Ji Zhang and Sanjiv Singh. Low-drift and real-time lidar odometry and mapping.

Autonomous Robots, 41(2):401–416, 2017. ISSN 15737527. doi: 10.1007/s10514-016-9548-2. 2.3.1