Tree Species Classification with High Resolution Satellite Imagery in a Multi-Label Setting

Irfan Hamid

Master of Science
Artificial Intelligence
School of Informatics
University of Edinburgh
2024

Abstract

Forests play a critical role in maintaining ecological balance, providing numerous ecological services such as carbon sequestration, biodiversity preservation, water cycle regulation, and soil stabilization. Given these vital functions, precise and efficient forest management practices are essential. To support such practices, this project explores the use of high-resolution satellite imagery combined with deep learning models to classify tree species in the Forest of Dean within a multi-label setting. Convolutional Neural Networks (CNNs) such as ResNet-34 and DenseNet-40 were utilized to perform the multi-label classification, successfully identifying multiple tree species within individual image patches. The classification accuracy for different tree species was then analyzed, and a comparative evaluation of the models was conducted. To address the challenge of high annotation costs for multi-label datasets, the Single Positive Multi-Label Learning (SPMLL) approach was implemented. SPMLL models, trained with only one positive label and no confirmed negatives per data instance, achieved performance metrics competitive with fully labeled supervised models, demonstrating the potential to significantly reduce annotation efforts while maintaining comparable classification accuracy. The results validate the practical utility and efficiency of combining satellite imagery with deep learning techniques for tree species detection, while the integration of the SPMLL approach provides a cost-effective method for tree species classification.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Irfan Hamid*)

Acknowledgements

Firstly, I would like to express my deepest gratitude to my supervisor, Dr. Michael Herrmann, for his guidance and support throughout this project. His insightful advice and encouragement have been instrumental in the successful completion of this work.

I am also deeply thankful to Dr. Juan C. Suárez and Dr. Miguel Ibañez Alvarez from Forest Research, Northern Research Station, Scotland, United Kingdom. This project was conducted in cooperation with them, and I am profoundly grateful for their guidance, detailed explanations on various topics related to forestry and for providing the essential datasets, including the Planet Labs SuperDove satellite imagery of the Forest of Dean and the Sub-Compartment Database (SCDB). Every visit to the Northern Research Station was a valuable learning experience and greatly contributed to the progress of this project.

I would also like to thank Shree Sonar, who is working on a similar project, for our collaborative discussions on the provided datasets. Together, we identified that a multi-label classification setting would be well-suited to the data. Following this, we each independently explored different approaches and methodologies within that setting. These initial discussions were instrumental in shaping the direction of this work.

Furthermore, I am deeply grateful to my mother and grandfather. During my time studying abroad, they have always provided me with unwavering support.

Finally, I would like to thank my friends for making this year in Edinburgh an unforgettable experience.

Table of Contents

Chapter 1

Introduction

1.1 Motivation

Forests are a vital and irreplaceable resource integral to ecological health, playing a crucial role in climate regulation [1], oxygen production, providing habitats for diverse species, and supporting biodiversity [35]. They also contribute significantly to water cycle regulation [8] and soil preservation, thereby enhancing environmental stability and human well-being [37]. The fundamental unit for forest inventories is the forest stand, which represents a large area of uniform tree species composition [6]. Accurate and timely identification of forest stand types and tree species is crucial for better understanding, managing, and preserving forests [50]. This identification supports effective forest management through precise tree species classification and growing stock volume estimation, which are critical for assessing forest resources [38] and also play a key role in evaluating carbon sequestration capacity [30]. Additionally, individual tree species identification is vital for forest inventory and monitoring [40]. The creation of these comprehensive inventories is instrumental in resource management, biodiversity assessment, and conservation efforts [51, 10].

In traditional forest surveys, determining the distribution of different tree species and forest stands involved extensive fieldwork which involved creating stand maps using traditional land survey techniques, which is both time-consuming and labor-intensive [42]. Compared to the traditional methods, there are simpler and more cost-effective approaches for detecting tree species, addressing an ongoing need in forest management [18]. Remote sensing technology offers a more efficient solution by gathering forest data over extensive areas, even in dense and hard-to-reach locations. Spectral images from remote sensing systems provide a practical and cost-effective

means to map tree species distributions [18, 16], thereby reducing the need for extensive fieldwork [4, 43]. By leveraging the unique spectral characteristics of different tree species [21], it is possible to extract species distribution information from these images. Spectral images encompass both multispectral and hyperspectral varieties and recent advancements in these imaging methods have increased their application in tree species detection studies.

Building on these advancements, deep learning (DL) techniques have been increasingly integrated with multispectral satellite images to further enhance tree species detection and classification. By applying convolutional neural networks (CNNs) [28] to the rich spectral data provided by multispectral images, researchers can accurately identify and map tree species across vast forested areas [52, 13, 29]. This integration allows for the automatic extraction of spectral and spatial features, significantly improving the precision and efficiency of forest monitoring and management [34, 31]. While these advancements have primarily focused on enhancing classification accuracy, it's important to note that much of this research has traditionally been conducted in a multi-class setting, where each image is assigned a single class from multiple distinct categories. However, real-world scenarios often involve more complexity, where images frequently contain multiple objects or, in the case of satellite images, various tree species simultaneously.

Multi-label classification is a machine learning approach that assigns multiple labels to each data instance simultaneously [58, 56, 32]. This differs from multi-class classification, where each data instance is assigned only one label out of many possible classes. In ecological studies, multi-label classification is particularly useful because it allows for the identification of multiple tree species within a single area in a satellite image. This approach captures the complexity of forest ecosystems more accurately by accommodating multiple labels for each image segment. However, obtaining multiple ground truth labels for multi-label classification is a significantly challenging task. It requires expertise, is time-consuming, and can be expensive, often necessitating specialists for accurate annotation [7]. In contrast to the multi-class scenario, which requires only one positive label per instance, multi-label classification demands a complete set of positive labels for each instance. To address this challenge, the Single Positive Multi-Label Learning (SPMLL) approach [5] significantly reduces annotation costs by requiring only one relevant label per example. By leveraging SPMLL methods combined with the learning capabilities of deep neural networks (DNNs), recent studies have empirically demonstrated that SPMLL can effectively lower annotation costs

while maintaining comparable performance in practice to the fully labeled case in some scenarios [5]. Compared to fully labeled datasets, SPMLL methods on single-positive labeled examples incur only a slight drop in performance, yet they drastically reduce the amount of supervision required to train multi-label classifiers in some cases.

1.2 Aims and Objectives

The aim of this project is to classify different tree species in the Forest of Dean using high-resolution satellite imagery from Planet Labs' SuperDove 8 CubeSats within a multi-label classification setting. To achieve this, labels for different tree species within the Forest of Dean sourced from the Sub-Compartment Database (SCDB) will be utilized. The classification will be performed using Convolutional Neural Networks (CNNs).

The objectives are:

1. First, utilize the high-resolution multispectral images from Planet Labs' SuperDove 8 CubeSats and implement a multi-label classification approach using CNNs to classify tree species with the fully labeled SCDB.

2. Next, apply the single-positive multi-label learning (SPMLL) approach to the same SCDB and validate this approach by comparing its performance against the fully labeled approach.

The hypothesis is that the single-positive multi-label learning (SPMLL) approach can achieve performance close to that of the fully labeled case with significantly reduced annotation for the SCDB. This would save substantial annotation effort for forest department officials while maintaining comparable classification accuracy. By aiming to reduce the annotation workload for further forestry applications, the SPMLL approach is expected to prove itself as a practical and efficient alternative for forest management.

1.3 Results Achieved

The experimental results of this project demonstrate the feasibility of using high-resolution satellite imagery for the multi-label classification of tree species in the Forest of Dean. By utilizing deep learning models such as ResNet-34 and DenseNet-40,

multiple tree species were successfully identified within individual image patches. To further validate these findings, a comparative evaluation of the models was conducted, highlighting their effectiveness in handling complex, multi-label classification tasks. Both ResNet-34 and DenseNet-40 delivered strong performance across various metrics. While ResNet-34 performed slightly better in the evaluated classification metrics overall, DenseNet-40 showed better performance for less abundant species when evaluated across a range of thresholds.

Additionally, the classification accuracy for various tree species was analyzed. The analysis revealed that Oak had the highest classification metrics. On the other hand, Corsican pine and Scots pine had the lowest metrics, reflecting the challenges in distinguishing these species due to their similar spectral features.

Furthermore, the implementation of the Single Positive Multi-Label Learning (SPMLL) approach, which involves training models with only one positive label and no confirmed negatives per data instance, demonstrated performance metrics that were competitive with those of fully labeled supervised models. This underscores the efficiency of the SPMLL approach in leveraging limited supervision to achieve robust classification performance. The findings suggest that SPMLL can substantially reduce the annotation burden while making a small trade-off in classification accuracy for the SCDB, making it a practical and cost-effective solution for forest management and further forestry application. By minimizing the need for extensive manual labeling, the SPMLL approach not only saves time and resources but also enhances the scalability of tree species classification using satellite imagery.

1.4 Structure of Dissertation

This dissertation is structured into the following chapters:

1. **Background**: This chapter covers the theoretical foundations and literature review relevant to this project. It discusses the concepts of Convolutional Neural Networks (CNNs), Residual Networks (ResNets), and Dense Convolutional Networks (DenseNets), as well as multi-label classification, including its challenges and specific techniques such as Single Positive Multi-Label Learning (SPMLL).

2. **Methodology**: This chapter describes the study area, the datasets used, and the data pre-processing steps. It also details the problem setup for both multi-label classification using the fully labeled dataset and the Single Positive Multi-

Label Learning (SPMLL) approach, as well as the experimental environment and settings.

3. **Evaluation Metrics**: This chapter describes the metrics used to evaluate the performance of the multi-label classification models.

4. **Results and Discussion**: This chapter presents the experimental results, comparing the performance of the ResNet-34 and DenseNet-40 models on fully labeled datasets, followed by an evaluation of these models when trained under the SPMLL approach against their fully labeled counterparts. The chapter concludes with a detailed classification accuracy analysis of the individual tree species.

5. **Conclusion and Future Work**: This chapter summarizes the key findings of the study and discusses the effectiveness of the methodologies used and outlines potential future research directions based on the project's findings.

Chapter 2

Background

2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) [28] have significantly advanced the field of computer vision. These architectures are known for their proficiency in tasks such as image recognition, object detection, and segmentation, CNNs excel by effectively learning and extracting complex hierarchical features from input images [27].

These networks are designed to process data with grid-like structures, such as images [14]. Unlike traditional fully connected neural networks that rely on general matrix multiplication, CNNs incorporate convolution operations in at least one of their layers [14].

A CNN typically consists of three main layer types: convolutional, pooling, and fully connected layers. The convolutional layers are responsible for producing feature maps. Filters or kernels slide across the image during this process, executing the convolution operation, expressed as $\mathbf{h} = f(\mathbf{x} \otimes \mathbf{w})$, where f is the activation function. These layers help identify diverse features such as edges and textures within the image [24]. Each individual filter produces a distinct feature map, revealing various attributes across the image's spatial domain [57]. The unique set of weights for each filter is applied across different regions of the image, allowing the network to identify similar features regardless of their location, thereby supporting translation invariance [20].

Neurons in convolutional layers of CNNs are sensitive to local receptive fields, which correspond to particular regions of the input image to which each neuron is responsive [28]. Unlike neurons in fully connected layers that connect to every input, neurons in convolutional layers only connect to these localized receptive fields. This selective connectivity allows CNNs to capitalize on the spatial structure of images,

enhancing both feature detection and the network's ability to generalize to new data [28].

Pooling layers follow the convolutional layers and work to reduce the spatial dimensions of the feature maps while retaining vital information [47]. The most common pooling technique is max pooling, which involves selecting the highest value from each local region of the feature map, effectively highlighting significant features within that region [15]. Another approach, average pooling, computes the average value across a region [48].

This layered processing extracts features from the input, which are then handled by fully connected layers to produce the final classification. Figure 2.1 represents a typical CNN architecture, where an input image undergoes multiple stages of convolution and pooling, refining and reducing the feature maps before they are sent to fully connected layers for the classification output.

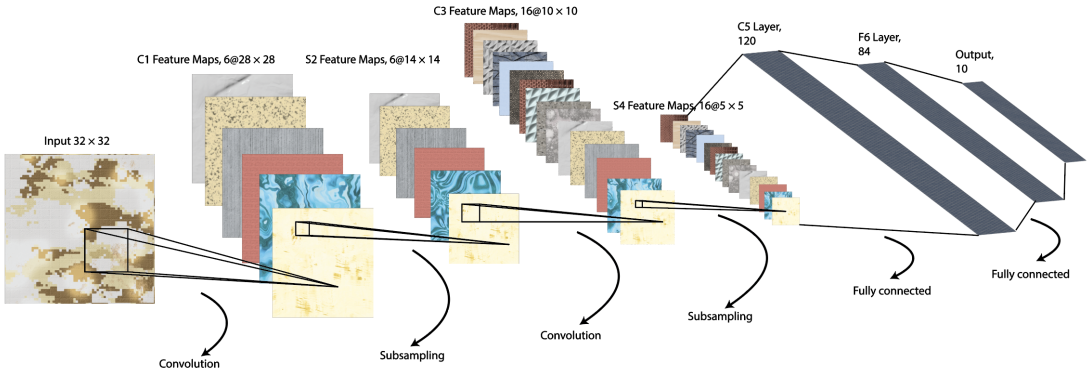

Figure 2.1: Architecture of a Convolutional Neural Network. (Figure created for this project using Adobe Illustrator.)

2.2 Residual Networks (ResNets)

The pursuit of developing more effective neural networks naturally raises the question: Is merely increasing the number of layers the key to improvement? However, this approach faces significant hurdles, primarily due to the well-known vanishing/exploding gradient problem [11]. This issue arises during the training of deep neural networks, where gradients can diminish or explode exponentially as they propagate backward through the network's layers. This phenomenon is closely related to how gradients are calculated during the backward pass of the backpropagation algorithm [45], which is

crucial for updating the network's weights. The backward pass involves computing the gradient of the loss function $\mathcal{L}$ with respect to the weights of each layer, as described by the equation:

$$\frac{\partial \mathcal{L}}{\partial W^{(l)}} = \frac{\partial \mathcal{L}}{\partial \mathbf{x}^{(L)}} \frac{\partial \mathbf{x}^{(L)}}{\partial \mathbf{x}^{(L-1)}} \cdots \frac{\partial \mathbf{x}^{(l+1)}}{\partial \mathbf{x}^{(l)}} \frac{\partial \mathbf{x}^{(l)}}{\partial W^{(l)}}. \tag{2.1}$$

In this expression, certain partial gradient terms may significantly differ in magnitude, being much larger or smaller than 1. The repeated multiplication of these terms can either cause the gradients to become exceedingly small, approaching zero, or excessively large, approaching infinity. Such extreme values of gradients hinder the network's ability to learn effectively by disrupting the proper adjustment of its weights, thereby impairing the overall learning process. Even when deeper networks begin to show signs of convergence, another problem emerges: the degradation of accuracy. As the network depth increases, accuracy initially plateaus and then experiences a steep decline. This drop in performance is not a result of overfitting; instead, it stems from the fact that adding more layers to an already deep network often leads to higher training errors [49].

Residual Networks (ResNets) [15] offer a solution to the vanishing gradient and the degradation problem by introducing "shortcut connections" that bypass some of the layers in the network. These connections allow the network to retain the ability to learn even as its depth increases. In a residual block, a convolutional layer (or a sequence of convolutions) is coupled with an identity mapping. Let $G^{(l)}$ represent the transformation applied by the l^{th} block. The output of this block, $\mathbf{h}^{(l)}$, is obtained by adding the input feature map $\mathbf{h}^{(l-1)}$ to the transformed output $G(\mathbf{h}^{(l-1)})$, resulting in the expression $\mathbf{h}^{(l)} = \mathbf{h}^{(l-1)} + G(\mathbf{h}^{(l-1)})$.

When residual blocks are stacked within a deep network, they create dual pathways for the input of each block: one through the convolutional layers and the other directly to the subsequent layer. These shortcut connections are key to preserving the gradients during backpropagation, as explained by:

$$\frac{\partial \mathbf{h}^{(l)}}{\partial \mathbf{h}^{(l-1)}} = \mathbb{1} + \frac{\partial G(\mathbf{h}^{(l-1)})}{\partial \mathbf{h}^{(l-1)}}. \tag{2.2}$$

The term $\mathbb{1}$ ensures that the gradients do not vanish, thereby enabling the network to train effectively even as more layers are added.

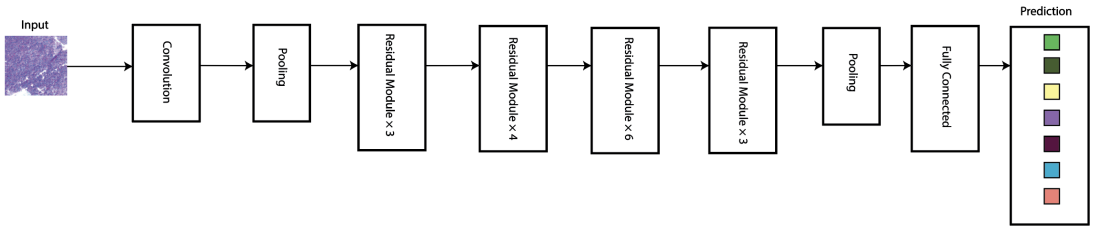

Figure 2.2: Architecture of a ResNet Model. (Figure created for this project using Adobe Illustrator.)

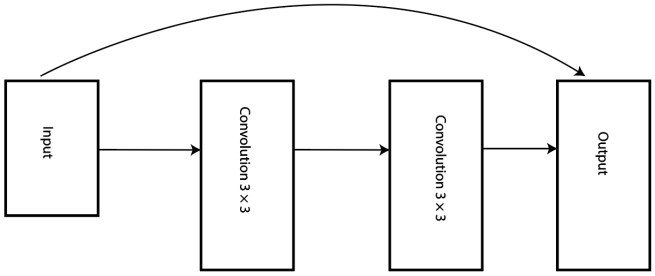

Figure 2.3: Structure of the Residual Module. (Figure created for this project using Adobe Illustrator.)

2.3 Dense Convolutional Networks (DenseNets)

Dense Convolutional Network (DenseNet) [17] is a deep learning architecture that has garnered significant attention for its innovative connectivity pattern, which facilitates improved information flow and gradient propagation throughout the network.

DenseNet introduces direct connections between any two layers, In contrast, in a traditional convolutional neural network, each layer receives input from the previous layer and passes its output to the next. DenseNet deviates from this pattern by ensuring that each layer obtains additional inputs from all preceding layers and passes on its feature maps to all subsequent layers, effectively creating $\frac{L(L+1)}{2}$ connections, where L is the depth of the network. This dense connectivity is expressed as:

$$\mathbf{h}_l = H_l([\mathbf{h}_0, \mathbf{h}_1, ..., \mathbf{h}_{l-1}]), \tag{2.3}$$

where $\mathbf{h}_l$ is the output of the l-th layer, H_l represents a composite function of

operations such as batch normalization [19], ReLU activation [12], and convolution, and $[\mathbf{h}_0, \mathbf{h}_1, ..., \mathbf{h}_{l-1}]$ denotes the concatenation of feature maps from layers 0 to $l-1$ [17]. This method of concatenation ensures that the l-th layer has direct access to the gradients and feature maps from all preceding layers, allowing for more efficient training and better feature reuse [17].

A key hyperparameter in DenseNet is the growth rate [17], denoted by k. The growth rate determines the number of feature maps added by each layer. Specifically, each layer produces k new feature maps, which are then concatenated with the existing feature maps from the previous layers. Therefore, the number of input feature maps for layer l is $k_0 + k \times (l-1)$, where k_0 represents the number of input channels [17].

DenseNet's architecture thus creates a global state within the network where each layer has access to the collective knowledge accumulated by all preceding layers. This global state allows the network to learn more robust features through feature reuse, as each layer can directly build upon the features learned by earlier layers [17].

One of the notable advantages of DenseNet's architecture is its ability to mitigate the vanishing gradient problem, DenseNet's direct connections between early and later layers enable more straightforward gradient flow, effectively preserving the gradient magnitude even in very deep networks. This enhanced gradient propagation not only accelerates the training process but also allows for the training of deeper models without encountering the typical difficulties associated with vanishing gradients.

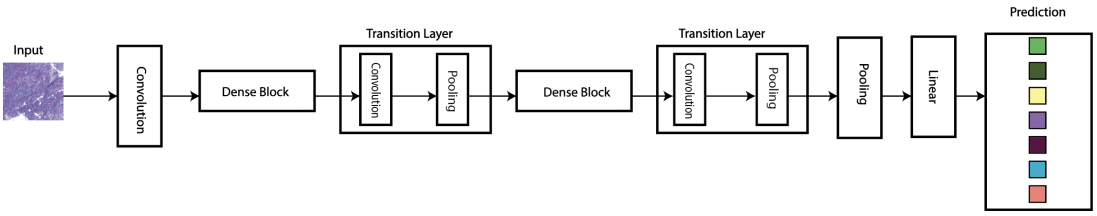

Figure 2.4: Architecture of a DenseNet Model. (Figure created for this project using Adobe Illustrator.)

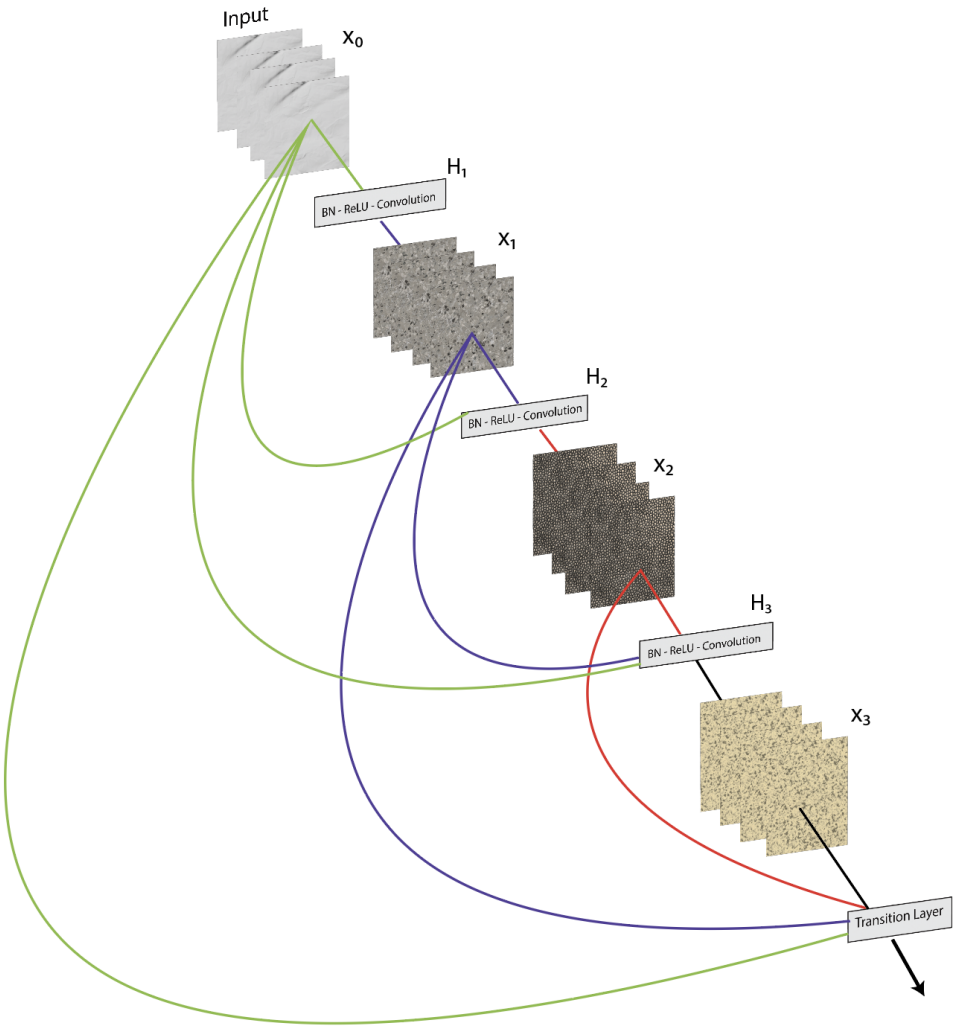

Figure 2.5: A dense block consisting of 4 layers with a growth rate of $k = 4$. Each layer uses the feature maps from all previous layers as input. (Figure created for this project using Adobe Illustrator.)

2.4 Multi-label Classification

The multi-class setting is a well-explored area in machine learning where each distinct object, represented by a unique instance or feature vector, is linked to a single label. If $\mathcal{X}$ denotes the input space and $\mathcal{Y}$ the label space, the goal of multi-class learning is to derive a function $f : \mathcal{X} \to \mathcal{Y}$ from a training set $\{(\mathbf{x}_i, y_i) \mid 1 \leq i \leq n\}$. Here, $\mathbf{x}_i \in \mathcal{X}$ signifies the features of an object, and $y_i \in \mathcal{Y}$ represents its corresponding label that defines its semantics. The key assumption in a multi-class setting is that each example

corresponds to only one label, implying it has a unique semantic meaning [58].

While traditional multi-class learning is prevalent and successful, it frequently falls short in real-world scenarios where objects possess multiple semantic meanings at the same time. For instance, a single satellite image might capture different types of trees such as oak, beech, and birch. In text categorization, a document could cover various topics like healthcare, technology, and politics. In healthcare, a medical image might indicate multiple conditions like fractures, tumors, and infections. Similarly, in retail, a product description might include multiple categories such as electronics, accessories, and home appliances.

To handle the complexity of objects that can belong to multiple categories, a practical approach is to assign a set of labels to each object. This method has given rise to multi-label learning. In contrast to traditional supervised learning where each object is associated with a single label, multi-label learning associates each object with multiple labels. The objective is to develop a model capable of predicting the appropriate set of labels for new, unseen instances [58, 56, 32]. Multi-label classification has applications across various domains such as natural language processing [21, 22, 44], information retrieval [41], bioinformatics [3] and audio classification [55, 2, 33].

However, acquiring ground truth labels in multi-label learning is challenging due to the labor-intensive nature of annotating each instance with multiple labels, the need for domain-specific expertise, and the risk of incomplete labeling, leading to false negatives [54, 53]. Furthermore, it is often more difficult to confirm the absence of certain labels than their presence [54].These factors make the annotation process time-consuming and expensive, which can limit the scalability and accuracy of multi-label learning models.

2.5 Single Positive Multi-Label Learning (SPMLL)

To tackle the challenge of high annotation costs in multi-label learning, the single-positive multi-label learning (SPMLL) [5] approach offers an efficient alternative. Instead of requiring annotators to label all possible categories for each instance, SPMLL asks for only one relevant label per training example. This approach significantly cuts down the labor and cost involved in the annotation process. This methodology is particularly beneficial in contexts where an image or data point may contain multiple objects or categories, but annotators only need to identify one. Similar challenges are present in non-vision fields like species distribution modeling [39], where the data

records positive observations without noting absences. The advantage of the single-positive setting is clear, it is often easier and more efficient for annotators to mark the presence of a specific class rather than confirm the absence of many others. This streamlined approach reduces the complexity and effort required for comprehensive labeling.

Empirical studies have demonstrated that integrating SPMLL with deep neural networks (DNNs) maintains comparable performance to fully labeled classifiers in some cases even with reduced annotation input. While there is a performance drop compared to fully labeled datasets, this is acceptable given the substantial reduction in annotation workload. Therefore, SPMLL appears to be a practical and cost-effective solution for training multi-label classifiers, offering robust model performance with significantly reduced annotation requirements in certain scenarios.

Chapter 3

Methodology

3.1 Area of Study

The area of study is the Forest of Dean, an ancient royal forest. The Forest is located between the Rivers Severn and Wye, near the border of England and Wales. It is a significant region for wildlife, featuring extensive woodlands and open spaces that create a diverse range of habitats, supporting a wide variety of species. The Forest has been utilized for both its timber and its abundant mineral resources, including iron ore, sandstone, and coal [9]. The geographical coordinates of the Forest of Dean are approximately 51.7891° N latitude and 2.5432° W longitude, and it was selected for this project as a result of the collaboration with Forest Research.

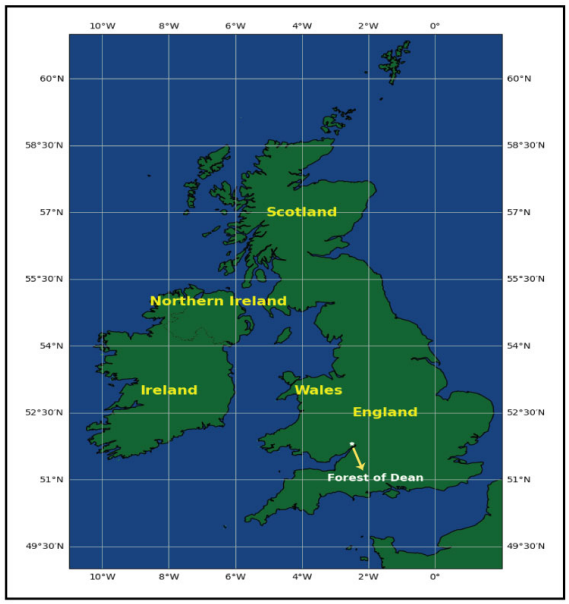

Figure 3.1: Location of the study area. (Figure generated for this project using Python.)

3.2 Dataset

3.2.1 Planet Labs SuperDove Satellite Imagery

Launched as part of the PlanetScope constellation, Planet Labs' SuperDove 8 CubeSats provide high-resolution satellite imagery essential for this study. These satellites, part of the continuous deployment by Planet Labs since 2020 are equipped with advanced imaging capabilities, capturing daily images of the Earth's surface [25],.

The SuperDove satellites, developed by Planet Labs, are small satellites known as CubeSats. Each SuperDove follows the CubeSat 3U form factor, which is a composite of three cube-like units. In this form factor, each "U" corresponds to a cube with dimensions of 10 cm x 10 cm x 10 cm. Therefore, a CubeSat 3U consists of three of these units stacked together, resulting in overall dimensions of 10 cm x 10 cm x 30 cm. These satellites can capture high-frequency images with a spatial resolution of about 3 meters per pixel, providing a detailed view necessary for ecological and environmental analysis [25]. The SuperDove satellites also feature a multi-spectral sensor that captures data across 8 distinct spectral bands. Table 3.1 summarizes these bands and their corresponding wavelength ranges [26]:

Band	**Wavelength Range (nm)**
Coastal Blue	431-452
Blue	465-515
Green I	513-549
Green	547-583
Yellow	600-620
Red	650-680
Red Edge	697-713
Near Infrared (NIR)	845-885

Table 3.1: Spectral Bands and Wavelength Ranges for SuperDove Satellites.

By leveraging the diverse spectral information captured across the 8 bands, deep learning methods can discern subtle differences in spectral signatures that are indicative of various tree species.

Figure 3.2: SuperDove satellite image of the Forest of Dean, captured on June 14, 2023. The image shows the area during the summer and is composed of the red, green, and blue bands. (Data provided by Forest Research, originally sourced from Planet Labs.)

3.2.2 Tree Species labels from the Sub-Compartment Database (SCDB)

Labels for trees within the Forest of Dean are sourced from the Sub-Compartment Database (SCDB). The SCDB is a vital geospatial data source managed by the Forestry Commission, which provides detailed information on the boundaries, species composition, and various attributes of forest stands. This database supports recording, monitoring, analysis, and reporting of forest information, which is crucial for effective forest management.

The SCDB operates as a vector data system, representing spatial features using geometric shapes such as points, lines, and polygons. In this context, sub-compartments within the SCDB are depicted as polygons that define the boundaries and areas of different sections within the forest. Each polygon (sub-compartment) is labeled with attributes such as species type, planting year, habitat types, and management history, allowing for a comprehensive understanding of each forest stand.

Integration with Forester GIS ensures synchronization between spatial data and inventory data within the SCDB, facilitating accurate mapping and analysis of forest resources. The SCDB's polygons clearly outline the boundaries of each forest stand and label them with the species of trees they contain.

Different colors are used to indicate various tree species in the vector data, as shown in Figure 3.3. This color-coding provides a quick visual assessment of species distribution across the forest. The primary tree species in the Forest of Dean are Beech, Birch, Corsican pine, Douglas fir, Norway spruce, Oak, Scots pine and Sweet chestnut, which outlines the main species targeted for the classification.

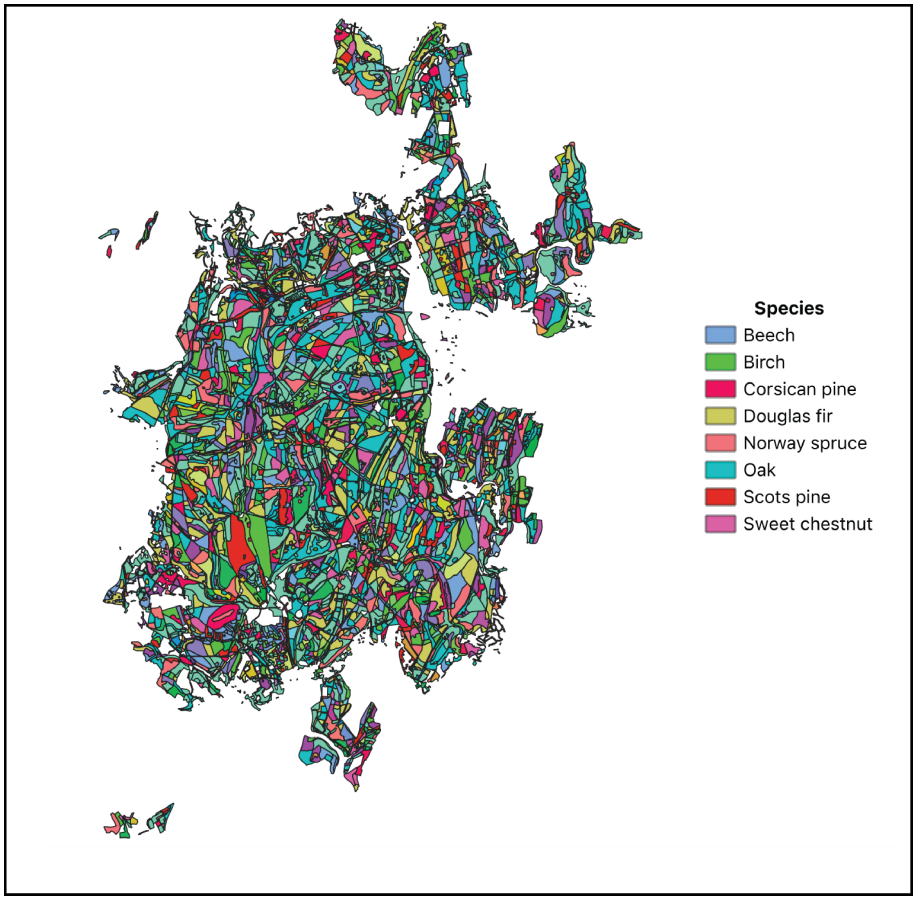

Figure 3.3: Sub-Compartment Database (SCDB) vector data indicating different tree species within the Forest of Dean. Each polygon represents a forest stand and is labeled with the tree species it contains, represented by different colors. (Data provided by Forest Research and Figure created for this project using QGIS software.)

3.3 Pre-processing the Data

3.3.1 Overlaying Vector Data on Satellite Image

The integration of SCDB vector data with satellite imagery is a crucial step in visualizing and analyzing spatial relationships within the Forest of Dean. SCDB vector data representing different tree species is overlaid onto high-resolution satellite images captured by the SuperDove satellites. This overlay process enables the visualization of how different tree species, represented by various colors, align with the raw satellite imagery, providing a comprehensive view of the forest's composition.

By overlaying the SCDB vector data on the satellite images, we can effectively assess the spatial distribution and density of different tree species within the forest. Figures A.1 and A.2 in the appendix further illustrate the spatial distribution of individual species, highlighting the varying density and presence of each species across the landscape.

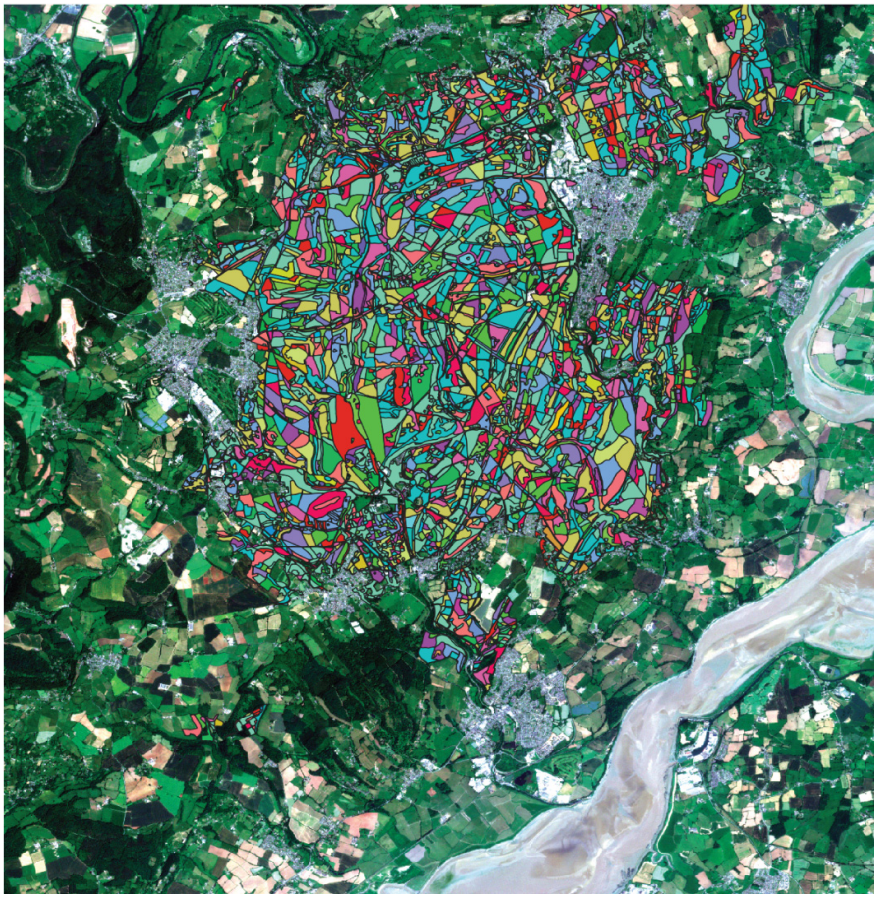

Figure 3.4: Overlaying SCDB Vector Data on the Satellite Image. (Figure created for this project using QGIS software.)

3.3.2 Extracting Patches and Corresponding Labels

The satellite image is divided into smaller patches to facilitate detailed analysis. Each patch is then examined to identify intersecting SCDB vector data features, specifically the tree species labels. These labels are assigned to the patches based on the intersection of the SCDB vector data with the satellite image. Each patch may have multiple labels corresponding to the sub-compartment (polygon) it encapsulates. For instance, if a patch covers three polygons, each representing a different tree species, the patch will be labeled with these three species from the intersecting polygons. This process creates a multi-label learning problem, where each patch can be classified into several categories simultaneously.

Figure 3.5 illustrates the patch extraction process, showing patches with their intersecting forest stands as defined by the SCDB vector data. This visualization helps in understanding how the patches correspond to different tree species within the forest (Patch extraction is done from the raw satellite data; it is visualized as shown in Figure 3.5 for better representation of the patches with their intersecting forest stands).

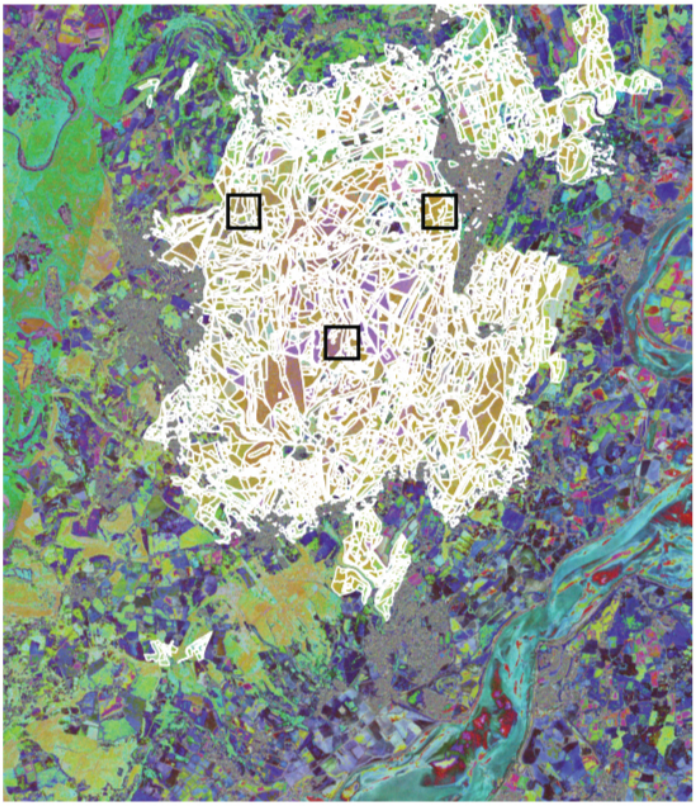

Figure 3.5: Visualization of patch extraction from satellite imagery with intersecting SCDB vector data, illustrating how each patch is assigned multiple labels based on the intersecting polygons. (Figure generated for this project using Python.)

The process of extracting patches with corresponding tree species labels is replicated across the entire area of interest (AOI). Each patch, once labeled, represents a sample that captures specific characteristics of the polygon it intersects. By aggregating these labeled patches from all parts of the AOI, a fully labeled multi-label dataset, denoted as $\mathcal{D}_{\text{FullyLabeled}}$, is created which is used for subsequent analysis and modeling.

The entire process of extracting and labeling the patches is summarized in Figure 3.6.

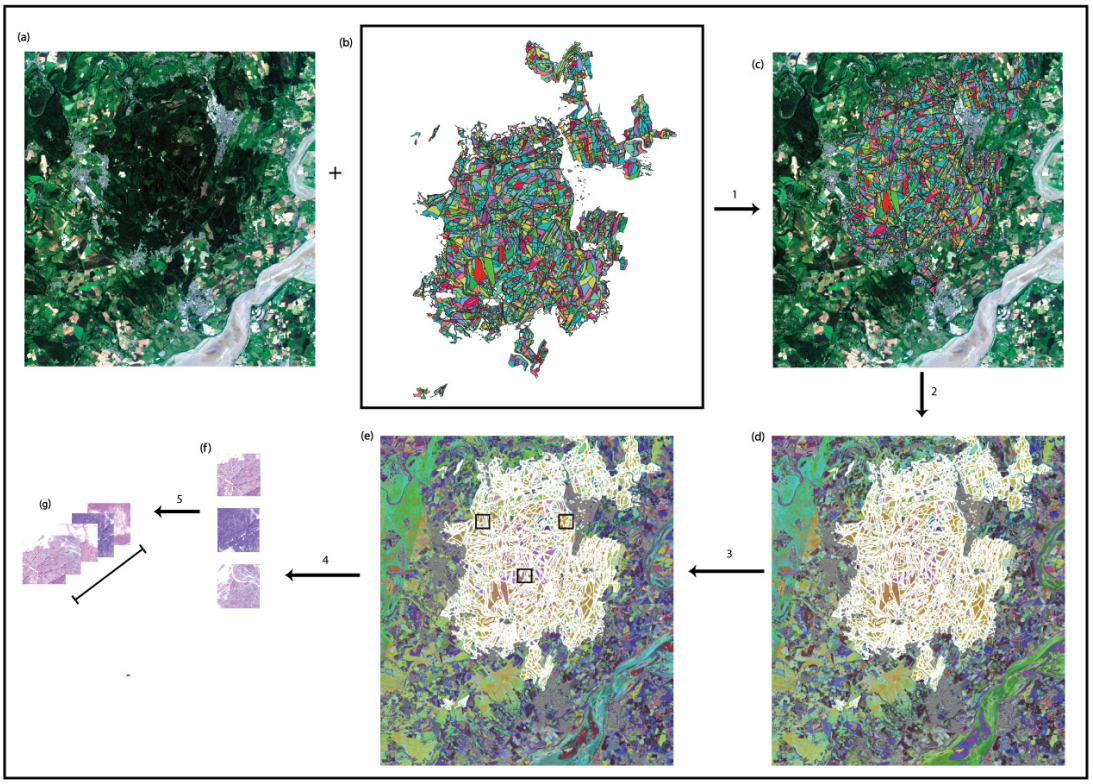

Figure 3.6: Overview of the entire process for extracting and labeling patches: (a) Input Satellite Image; (b) SCDB Vector Data; (c) Overlaying SCDB Vector Data on Satellite Image; (d) and (e) Extracting Patches and corresponding Labels; (f) Visualizing Patches; (g) Creating a collection of training and test samples. (Figure created for this project using Adobe Illustrator.)

3.4 Problem Setup

Let $\mathcal{X} = \mathbb{R}^d$ be a d-dimensional input space and $\mathcal{Y}$ be a label space with labels $[y_1, y_2, \ldots, y_K]$, where K denotes the number of classes. In a standard multi-class classification problem, each input data image $\mathbf{x} \in \mathcal{X}$ is assigned a single label from

$\mathcal{Y}$. In contrast, in a multi-label classification setting, each $\mathbf{x} \in \mathcal{X}$ is associated with a vector of labels $\mathbf{t}$ in the label space $\mathcal{T} = \{0,1\}^K$. In this context, an entry $t_i = 1$ signifies that the i^{th} class is associated with $\mathbf{x}$, whereas $t_i = 0$ indicates that the i^{th} class is not associated with $\mathbf{x}$.

The objective is to determine a function $f : \mathcal{X} \to [0,1]^K$ that predicts the relevant labels for each $x \in \mathcal{X}$.

Let the set $\mathcal{F}$ represent our models and $\mathcal{Z}$ represent a domain. Let l be a function from $\mathcal{F} \times \mathcal{Z}$ to the set of nonnegative real numbers, $l : \mathcal{F} \times \mathcal{Z} \to \mathbb{R}^+$. Here, $\mathcal{Z} = \mathcal{X} \times \mathcal{T}$, and l is the multi-label loss function. The risk function, which is the expected loss of a classifier $f \in \mathcal{F}$ with respect to a probability distribution P over $\mathcal{Z}$, is defined as follows [46]:

$$L_P(f) = \mathbb{E}_{z \sim P}[l(f,z)]. \tag{3.1}$$

The formal goal in a multi-label classification problem is to find an f that minimizes the risk function $L_P(f)$.

3.4.1 Multi-label Classification Using The Fully Labelled Dataset

Let $\mathbf{t}^{(n)} = [t_1^{(n)}, t_2^{(n)}, \ldots, t_K^{(n)}]^\top \in \{0,1\}^K$ denote the true presence (1) or absence (0) of K different tree species for image $\mathbf{x}^{(n)}$, where $t_m^{(n)} = 1$ if the m-th label is associated with $\mathbf{x}^{(n)}$ and $t_m^{(n)} = 0$ if the m-th label is not associated with $\mathbf{x}^{(n)}$. The vector of predictions for an input $\mathbf{x}^{(n)}$ is denoted as $\hat{\mathbf{t}}^{(n)} \in \{0,1\}^K$, where $\hat{t}_m^{(n)}$ denotes the m-th entry of $\hat{\mathbf{t}}^{(n)}$.

The model generates an estimate of $\mathbf{t}$ for any image $\mathbf{x}$ within the input space $\mathcal{X}$, based on observed data from the training subset of $\mathcal{D}_{\text{FullyLabeled}}$, which is denoted as $\mathcal{D}_{\text{FullyLabeledTraining}} = \{(\mathbf{x}^{(n)}, \mathbf{t}^{(n)})\}_{i=1}^N$. The predictions will be given by $\hat{\mathbf{t}} = f_\theta(\mathbf{x})$, where f_θ is a CNN (Convolutional Neural Network) with parameters θ. The prediction $\hat{\mathbf{t}} \in [0,1]^K$ represents our estimate of how likely each tree species is present in image $\mathbf{x}$.

The loss function used to train the CNN models is the binary cross-entropy loss (BCE). For a data point $(\mathbf{x}^{(n)}, \mathbf{t}^{(n)})$, the BCE loss is described as the following:

$$L_{BCE}(\hat{\mathbf{t}}^{(n)}, \mathbf{t}^{(n)}) = -\frac{1}{K} \sum_{m=1}^{K} \left[t_m^{(n)} \cdot \log(\hat{t}_m^{(n)}) + (1 - t_m^{(n)}) \cdot \log(1 - \hat{t}_m^{(n)}) \right], \tag{3.2}$$

where $\mathbf{t}^{(n)}$ represents the fully observed label vector.

3.4.2 Single Positive Multi-label Learning Classification

The fully labeled multi-label dataset, $\mathcal{D}_{\text{FullyLabeled}}$, is corrupted by discarding annotations from its training subset, $\mathcal{D}_{\text{FullyLabeledTraining}}$. Specifically, single positive multi-label learning (SPMLL) is simulated by randomly selecting one positive label to retain for each training example, while the remaining labels are treated as unobserved. The test subset $\mathcal{D}_{\text{SpmllTest}}$, however, remains unaltered and is the same as $\mathcal{D}_{\text{FullyLabeledTest}}$, retaining its full labels for evaluation purposes.

The SPMLL training subset is defined as $\mathcal{D}_{\text{SpmllTraining}} = \{(\mathbf{x}^{(n)}, o^{(n)})\}_{i=1}^{N}$, where $o^{(n)} \in \mathcal{Y}$ represents the observed single-positive label for $\mathbf{x}^{(n)}$. It is important to note that $o^{(n)}$ belongs to the relevant label set $\mathcal{T}^{(n)}$, even though $\mathcal{T}^{(n)}$ is not directly observable.

For each SPMLL training instance $(\mathbf{x}^{(n)}, o^{(n)})$, a observed single-positive vector $\mathbf{s}^{(n)} = [s_1^{(n)}, s_2^{(n)}, \ldots, s_K^{(n)}] \in \{1, \varnothing\}^K$ indicates whether the m-th label is the observed positive label, such that $s_m^{(n)} = 1$ if $m = o^{(n)}$, and $s_m^{(n)} = \varnothing$ otherwise. $s_m^{(n)} = \varnothing$ implies that the corresponding $s_m^{(n)}$ is unobserved and could be either 0 or 1. The SPMLL training subset now can also be denoted as $\mathcal{D}_{\text{SpmllTraining}} = \{(\mathbf{x}^{(n)}, s^{(n)})\}_{i=1}^{N}$, where $\mathbf{s}^{(n)}$ is the observed single-positive vector. The aim of SPMLL is to develop a multi-label classifier $g : \mathcal{X} \to [0,1]^K$ using $\mathcal{D}_{\text{SpmllTraining}}$, enabling it to determine a set of relevant labels for a new, unseen instance.

(i) Assume Negatives

Now, for the unobserved labels in $[s_1^{(n)}, s_2^{(n)}, \ldots, s_K^{(n)}]$, we treat them as negative, meaning $s_m^{(n)} = 0$ if $s_m^{(n)} = \varnothing$. This approach leads to the "assume negative" loss, which is defined as follows [5]:

$$L_{AN}(\hat{\mathbf{t}}^{(n)}, \mathbf{s}^{(n)}) = -\frac{1}{K} \sum_{m=1}^{K} \left[s_m^{(n)} \cdot \log(\hat{t}_m^{(n)}) + (1 - s_m^{(n)}) \cdot \log(1 - \hat{t}_m^{(n)}) \right]. \tag{3.3}$$

The limitation of the assume negative approach is that it may lead to an increased number of false negatives.

(ii) Weak Assume Negatives

A method to reduce the impact of false negatives is to assign less weight to the terms in the loss function that correspond to negative labels. This can be achieved by introducing a parameter $\gamma \in [0,1]$. The resulting "weak assume negative" (WAN) loss is defined as follows [5]:

$$L_{WAN}(\hat{\mathbf{t}}^{(n)}, \mathbf{s}^{(n)}) = -\frac{1}{K} \sum_{m=1}^{K} \left[s_m^{(n)} \cdot \log(\hat{t}_m^{(n)}) + \gamma \cdot (1 - s_m^{(n)}) \cdot \log(1 - \hat{t}_m^{(n)}) \right]. \tag{3.4}$$

Notations	Mathematical Meanings
$\mathcal{X}$	d-dimensional input space ($\mathbb{R}^d$).
$\mathcal{Y}$	Label space with K possible class labels, $[y_1, y_2, \ldots, y_K]$.
$\mathcal{T}$	Label space of possible multi-label vectors. $\{0,1\}^K$
$\mathbf{x}$	d-dimensional input image feature vector, $[x_1, x_2, \ldots, x_d]$, $\mathbf{x} \in \mathcal{X}$.
$\mathbf{t}$	Fully observed label vector, $[t_1, t_2, \ldots, t_K] \in \{0,1\}^K$. Indicates the presence (1) or absence (0) of each label.
$\hat{\mathbf{t}}$	Predicted multi-label vector, $[\hat{t}_1, \hat{t}_2, \ldots, \hat{t}_K] \in \{0,1\}^K$. The model's prediction for the presence or absence of each label.
$\mathbf{s}$	Observed single-positive vector, $[s_1, s_2, \ldots, s_K] \in \{1, \varnothing\}^K$. Represents observed positive label and unobserved labels.
$\mathcal{D}_{\text{FullyLabeledTraining}}$	Fully labeled multi-label training subset, $\mathcal{D}_{\text{FullyLabeledTraining}} = \{(\mathbf{x}_i, \mathbf{t}_i)\}_{i=1}^N$. Training subset where each input image has a complete label vector.
$\mathcal{D}_{\text{SpmllTraining}}$	Single Positive Multi-Label Learning training subset, $\mathcal{D}_{\text{SpmllTraining}} = \{(\mathbf{x}_i, \mathbf{s}_i)\}_{i=1}^N$. Training subset where only one positive label and no confirmed negatives is observed per input image.
$\mathcal{L}_{\text{BCE}}$	Binary Cross-Entropy loss function
L_{WAN}	Weak Assume Negative loss function

Table 3.2: Notations and Their Mathematical Meanings.

3.5 Experiment Environment and Settings

In this project, four models were trained: ResNet-34 and DenseNet-40 architectures on $\mathcal{D}_{\text{FullyLabeledTraining}}$, and ResNet-34 and DenseNet-40 architectures on $\mathcal{D}_{\text{SpmllTraining}}$ (employing SPMLL). ResNet-34, with its 34 layers, and DenseNet-40, with its 40 layers, were chosen for their balance between performance and computational efficiency. These architectures are well-regarded for their ability to capture complex patterns in data while maintaining reasonable training times and resource requirements. Experiments were conducted on Google Colab using NVIDIA A100 Tensor Core GPUs with CUDA version 12.1.

For these four models, the Adam optimizer [23] was used to update the model parameters, with optimal base learning rates determined for each model through careful tuning during the training process. While the maximum training duration was 67 epochs, most models converged earlier and required fewer epochs to complete training. A batch size of 64 was used across the experiments.

Batch Normalization (BN) was applied in all models to stabilize and accelerate training by normalizing the inputs to the layers, which helps to reduce internal covariate shift and allows for more consistent activation distributions throughout the network [19]. The activation function employed across all models was the Rectified Linear Unit (ReLU) [36], defined as:

$$\text{ReLU}(x) = \begin{cases} x, & \text{if } x > 0, \\ 0, & \text{if } x \leq 0. \end{cases} \tag{3.5}$$

ReLU was chosen for its simplicity, which enables faster computation, as well as its ability to avoid the vanishing gradient problem, thereby facilitating more effective learning in deep networks. Additionally, ReLU contributes to efficient training by selectively activating only a subset of neurons, which introduces sparsity. This sparsity not only makes the models more computationally efficient but also often leads to better overall performance.

Chapter 4

Evaluation Metrics

The evaluation metrics used to assess the performance of the different multilabel classification models are Precision, Recall, F1 Score, and Mean Average Precision (mAP). Each of these metrics provides unique insights into the model's performance from different perspectives.

4.1 Precision

Precision is the ratio of correctly predicted positive instances to the total predicted positive instances. It quantifies the proportion of positive predictions made by the model that are actually correct.

For a single label, precision is calculated as:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}. \tag{4.1}$$

For a multilabel classification problem, precision is averaged across all labels using the following methods:

- **Micro-averaged Precision** aggregates the contributions of all labels to compute the average metric:

$$\text{Micro-averaged Precision} = \frac{\sum_{i=1}^{L} \text{TP}_i}{\sum_{i=1}^{L} (\text{TP}_i + \text{FP}_i)}. \tag{4.2}$$

- **Macro-averaged Precision** computes the metric independently for each label and then takes the average:

$$\text{Macro-averaged Precision} = \frac{1}{L} \sum_{i=1}^{L} \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}. \tag{4.3}$$

4.2 Recall

Recall is the ratio of correctly predicted positive instances to the total actual positive instances. It quantifies the proportion of actual positive instances that the model accurately identifies.

For a single label, recall is calculated as:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}. \tag{4.4}$$

For a multilabel classification problem, recall is also averaged using micro and macro averaging methods:

- **Micro-averaged Recall** aggregates the contributions of all labels to compute the average metric:

$$\text{Micro-averaged Recall} = \frac{\sum_{i=1}^{L} \text{TP}_i}{\sum_{i=1}^{L} (\text{TP}_i + \text{FN}_i)}. \tag{4.5}$$

- **Macro-averaged Recall** computes the metric independently for each label and then takes the average:

$$\text{Macro-averaged Recall} = \frac{1}{L} \sum_{i=1}^{L} \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}. \tag{4.6}$$

4.3 F1 Score

The F1 Score is the harmonic mean of precision and recall, providing a balance between the two metrics. It is particularly useful when balancing precision and recall.

For a single label, the F1 Score is calculated as:

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{4.7}$$

For a multilabel classification problem, the F1 Score is also averaged using micro and macro averaging methods:

- **Micro-averaged F1 Score** aggregates the contributions of all labels to compute the average metric:

$$\text{Micro-F1} = 2 \cdot \frac{\text{Micro-P} \cdot \text{Micro-R}}{\text{Micro-P} + \text{Micro-R}}, \tag{4.8}$$

  where Micro-F1 is the Micro-averaged F1 Score, Micro-P is the Micro-averaged Precision, and Micro-R is the Micro-averaged Recall.

- **Macro-averaged F1 Score** computes the metric independently for each label and then takes the average:

$$\text{Macro-averaged F1 Score} = \frac{1}{L}\sum_{i=1}^{L} 2 \cdot \frac{\text{Precision}_i \cdot \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i}. \tag{4.9}$$

4.4 Mean Average Precision (mAP)

Mean Average Precision (mAP) is a common metric used to evaluate the performance of classification and detection systems, particularly in multi-label settings. It provides a comprehensive measure of the model's ability to correctly predict multiple labels for each instance.

4.4.1 Average Precision (AP)

AP is a measure used to summarize the precision-recall curve into a single number. The precision-recall curve plots precision (P) against recall (R) at different threshold levels.

The formula for AP can be represented as:

$$\text{AP} = \sum_{n} (R_n - R_{n-1}) P_n, \tag{4.10}$$

where:

- P_n and R_n are the precision and recall at the n-th threshold.
- R_{n-1} is the recall at the previous threshold.

For each label, AP is calculated by plotting the precision-recall curve and taking the weighted sum of precisions at each threshold, where the weight is the increase in recall since the previous threshold.

4.4.2 Mean Average Precision (mAP)

Mean AP (mAP) is simply the mean of the AP values calculated for each label in a multi-label classification task. If there are L labels, then:

$$\text{mAP} = \frac{1}{L}\sum_{i=1}^{L} \text{AP}_i, \tag{4.11}$$

where AP_i is the average precision for the i-th label.

Chapter 5

Results and Discussions

5.1 Classification Results

This section presents the results of the multi-label classification experiments. The results are divided into two parts: results for fully labeled datasets and results for single positive multi-label learning. The Tables summarize various performance metrics, such as Precision, Recall, F1 Score, and Mean Average Precision (mAP), for different models and loss functions.

5.1.1 Classification Results for Fully Labeled Dataset

Model	Loss	Precision	Recall	F1 Score
ResNet-34	$\mathcal{L}_{\text{BCE}}$	74.16	74.76	74.46
DenseNet-40	$\mathcal{L}_{\text{BCE}}$	75.08	72.69	73.87

Table 5.1: Micro Average Evaluation Metrics (%).

Model	Loss	Precision	Recall	F1 Score
ResNet-34	$\mathcal{L}_{\text{BCE}}$	72.70	73.45	72.75
DenseNet-40	$\mathcal{L}_{\text{BCE}}$	72.25	69.46	69.63

Table 5.2: Macro Average Evaluation Metrics (%).

Model	Loss	mAP (%)
ResNet-34	$\mathcal{L}_{\text{BCE}}$	74.77
DenseNet-40	$\mathcal{L}_{\text{BCE}}$	74.81

Table 5.3: Mean Average Precision (mAP) (%).

5.1.2 Classification Results for Single Positive Multi-Label Learning

Model	Loss	Precision	Recall	F1 Score
ResNet-34 with SPMLL	$\mathcal{L}_{\text{WAN}}$	74.04	72.62	73.32
DenseNet-40 with SPMLL	$\mathcal{L}_{\text{WAN}}$	73.07	72.47	72.77

Table 5.4: Micro Average Evaluation Metrics (%).

Model	Loss	Precision	Recall	F1 Score
ResNet-34 with SPMLL	$\mathcal{L}_{\text{WAN}}$	70.44	69.09	67.18
DenseNet-40 with SPMLL	$\mathcal{L}_{\text{WAN}}$	69.44	68.73	66.85

Table 5.5: Macro Average Evaluation Metrics (%).

Model	Loss	mAP (%)
ResNet-34 with SPMLL	$\mathcal{L}_{\text{WAN}}$	68.96
DenseNet-40 with SPMLL	$\mathcal{L}_{\text{WAN}}$	69.75

Table 5.6: Mean Average Precision (mAP) (%).

5.2 Comparison between CNN Models trained on the Fully Labeled Dataset

The evaluation of ResNet-34 and DenseNet-40 models using mean average precision (mAP), micro, and macro metrics on the fully labeled dataset highlights subtle dif-

ferences and important observations about their performance on the given multi-label classification problem.

The mean average precision (mAP) scores were very close, with ResNet-34 at 74.77% and DenseNet-40 at 74.81%, as depicted in Table 5.3. Although DenseNet-40 shows a minutely higher mAP, the difference is negligible and may not be practically significant. mAP is a metric that evaluates the performance of a model across multiple thresholds. A model might be optimized to perform well at a certain threshold, achieving high precision and recall, but might not maintain this performance across a range of thresholds. Both ResNet-34 and DenseNet-40 demonstrate a strong ability to maintain high precision and recall across different thresholds, as evidenced by their mAP scores.

ResNet-34 achieved a micro-averaged precision of 74.16%, recall of 74.76%, and F1 score of 74.46%. DenseNet-40, on the other hand, performed with a micro-averaged precision of 75.08%, recall of 72.69%, and F1 score of 73.87%, as shown in Table 5.1. For macro average metrics, ResNet-34 showed a macro-averaged precision of 72.70%, recall of 73.45%, and F1 score of 72.75%. DenseNet-40's macro average metrics were slightly lower, with a macro-averaged precision of 72.25%, recall of 69.46%, and F1 score of 69.63%, as shown in Table 5.2.

While the micro scores for both models are very close, with ResNet-34 having a slight edge in classification performance, the macro scores show more noticeable differences. To understand these differences, it is important to consider the distribution of tree species within the dataset. Figure 5.4 shows the distribution of tree species by their area of coverage, while Figures A.1 and A.2 in the appendix illustrate the spatial distribution and density of the individual tree species. These figures highlight the abundance of species such as Beech, Birch, Douglas fir, and Oak, as well as the presence of less common species like Scots pine, Corsican pine, and Norway spruce. This context is crucial for interpreting the differences in both models' performance as macro scores are sensitive to the performance on less frequent classes.

Macro metrics are calculated at a single, fixed threshold and average the precision and recall values across all labels, treating each label equally, regardless of its frequency in the dataset. This makes macro metrics sensitive to the model's performance on less frequent classes. In this evaluation, ResNet-34 shows higher macro scores, indicating that it performs better for less frequent species.

However, the average precision (AP) scores, which evaluate model performance across various thresholds, present a different perspective. As shown in Figure 5.5, DenseNet-40 achieves higher AP scores for less frequent species compared to ResNet-

34. This suggests that DenseNet-40 is more effective at identifying true positives across varying thresholds for these less frequent species. This discrepancy arises because AP measures the area under the precision-recall curve and evaluates the model's ability to maintain high precision and recall across different thresholds. Thus, AP provides a comprehensive assessment of the model's performance by considering how well it ranks true positives higher than false positives across all possible thresholds. Therefore, when considered across multiple thresholds, DenseNet-40 performs better than ResNet-34 for less frequent species. However, when a fixed threshold is applied, as in macro calculations, ResNet-34 may perform better for these species.

Another important observation is that while DenseNet-40 is better at identifying less frequent species across various thresholds, ResNet-34 performs better for abundant species across the same range of thresholds, as shown in Figure 5.5. This balance results in nearly identical mAP scores for both models, highlighting that they each have their strengths in different aspects of classification performance. The mAP, which averages the AP scores across all labels, effectively captures this balance.

The differences in AP and macro metrics highlight the importance of considering multiple evaluation criteria when assessing model performance, as they can reveal different aspects of a model's strengths and weaknesses, especially in multi-label classification tasks involving species with varying frequencies.

5.3 Comparison of CNN Models Trained on Fully Labeled Dataset versus Single Positive Multi-Label Learning

Under SPMLL, ResNet-34 achieves a micro-averaged precision of 74.04%, recall of 72.62%, and F1 score of 73.32%. DenseNet-40 achieves a micro-averaged precision of 73.07%, recall of 72.47%, and F1 score of 72.77%, as illustrated in Table 5.4. Despite being trained with only one positive label and no confirmed negatives per instance, the micro performance of the ResNet-34 and DenseNet-40 models trained under SPMLL is remarkably close to that of their respective fully labeled counterparts. This highlights the efficiency of the SPMLL approach in maintaining high micro classification performance with significantly reduced labeling effort. It indicates that SPMLL can effectively leverage limited supervision to achieve results comparable to those obtained with extensive labeling for the SCDB, thereby reducing the annotation

burden without significant loss in performance.

Macro average metrics further demonstrate the robustness of SPMLL. Under SPMLL, ResNet-34 achieves a macro-averaged precision of 70.44%, recall of 69.09%, and F1 score of 67.18%, while DenseNet-40 achieves a macro-averaged precision of 69.44%, recall of 68.73%, and F1 score of 66.85%, as shown in Table 5.5. Although these results are expectedly lower than their fully labeled counterparts, they remain competitive with the performance of fully supervised models. In multi-label classification tasks, the ability to generalize well across all classes, particularly the less frequent ones, is crucial. Rare classes can significantly influence overall performance if they are not accurately identified. The fact that the macro metrics for SPMLL trained models are only slightly reduced compared to fully labeled training indicates that these models retain a strong ability to classify these less common species. This effective generalization across both common and rare classes demonstrates the potential of SPMLL to provide reliable classification results even with reduced supervision.

The mean average precision (mAP) scores under SPMLL are 68.96% for ResNet-34 and 69.75% for DenseNet-40, as shown in Table 5.6. The AP scores across various classes are illustrated in Figure 5.5. Though these scores are understandably lower than those achieved with fully labeled data, they still approach the performance levels of their full label supervision counterparts. The comptetive mAP scores indicate that SPMLL can maintain a high level of accuracy and reliability in predictions across a range of thresholds for the SCDB, even with the reduced supervision of using only a single positive label per instance.

Both ResNet-34 and DenseNet-40 models demonstrate that SPMLL can deliver competitive results which approaches the level of their fully labeled counterparts for the SCDB, despite the reduction in label information. This ability to perform well with minimal supervision underscores the robustness and efficiency of the SPMLL approach. It shows that the models can effectively learn and generalize from limited data, capturing the essential patterns and correlations needed for accurate classification. By significantly reducing the annotation workload without substantial loss in performance, SPMLL offers a practical and efficient solution to multi-label classification tasks, making it particularly valuable in scenarios where manual labeling is resource-intensive and costly.

Figure 5.1 shows the true labels and predicted labels for three test instances. Despite the SPMLL models being trained with only one positive label and no confirmed negatives per instance, they are able to predict and match multiple positive labels for

the SCDB. This indicates that SPMLL effectively recovers a significant number of the unlabeled positives, as evidenced by the predicted labels.

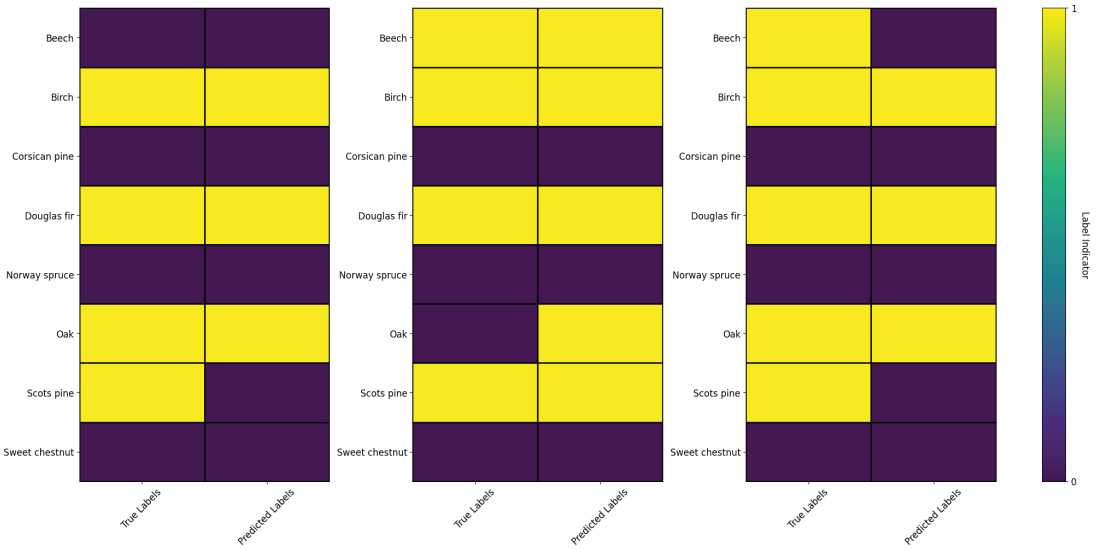

Figure 5.1: True vs Predicted labels for three test set instances from a model trained with SPMLL. Each subfigure shows the true labels on the left and the predicted labels on the right. The yellow color indicates the presence of a label, and the purple color indicates its absence.

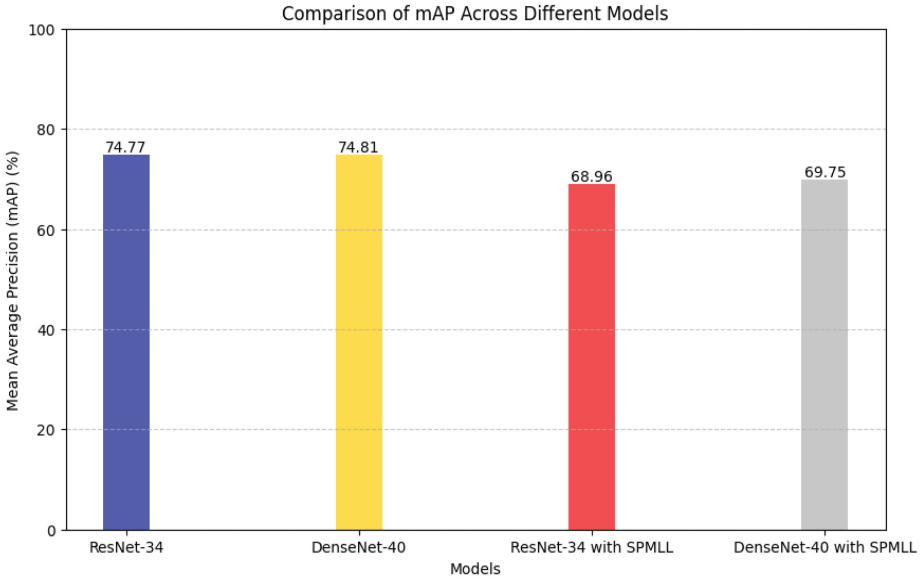

Figure 5.2: Comparison of Mean Average Precision (mAP) across different models. ”ResNet-34” and ”DenseNet-40” refer to models trained with fully labeled dataset, while ”ResNet-34 with SPMLL” and ”DenseNet-40 with SPMLL” refer to models trained with Single Positive Multi-Label Learning.

5.4 Classification Accuracy Analysis of the Individual Tree Species

Figure 5.5 show the average precision (AP) for various tree species across the four models: ResNet-34, DenseNet-40, and their SPMLL-trained counterparts. The AP scores reveal significant insights into the models' performance for each species.

The classification differences between the nine tree species are significantly influenced by their spectral signatures, with the most notable distinctions occurring in the red edge and NIR bands, particularly pronounced in the NIR band.

Oak, which covers the largest area (1049.52 ha), consistently achieved the highest AP scores across all model configurations. Although Oak's reflectance profile shares similarities with other species in several spectral bands, it exhibits a distinctive pattern in the NIR band, setting it apart from other species except for Sweet chestnut. This distinct NIR reflectance enables the models to accurately classify Oak, despite some overlap in other spectral bands.

Beech and Douglas fir, with area coverages of 838 ha and 880.36 ha respectively, also achieved high AP scores. Their reflectance profiles are unique not only compared to each other but also across the entire dataset, particularly in the NIR band. These distinct spectral characteristics contribute to the models' ability to effectively differentiate between these species.

Birch and Sweet chestnut, covering 576.39 ha and 575.20 ha respectively, attained relatively high AP scores. However, despite Sweet chestnut's similar area coverage to Birch, its lower AP scores can be attributed to its NIR reflectance being similar to that of Oak. While Sweet chestnut's NIR values closely resemble Oak's, its unique reflectance in other bands aids its classification.

Corsican pine and Scots pine displayed the lowest AP scores, corresponding to their limited area coverage, 231.01 ha for Corsican pine and 217.43 ha for Scots pine. The difficulty in distinguishing between these species is compounded by their highly similar reflectance profiles across all spectral bands, providing fewer distinct features for the models to learn from.

In contrast, Norway spruce, with an area coverage of 279.51 ha, showed better AP scores than Corsican pine and Scots pine. This improvement is likely due to its more distinct reflectance profile across all bands, which helps differentiate it from other species.

Overall, these findings demonstrate how the distribution of training instances and

the distinctiveness of spectral curves influence the models' classification performance. Species with abundant training instances and unique reflectance profiles, such as Oak, Beech, and Douglas fir, achieve higher AP scores, while those with fewer training instances and similar spectral patterns, like Corsican pine and Scots pine, present more challenges. Norway spruce, despite having area coverage similar to that of Corsican pine and Scots pine, achieved relatively better AP scores due to its distinctive spectral signatures.

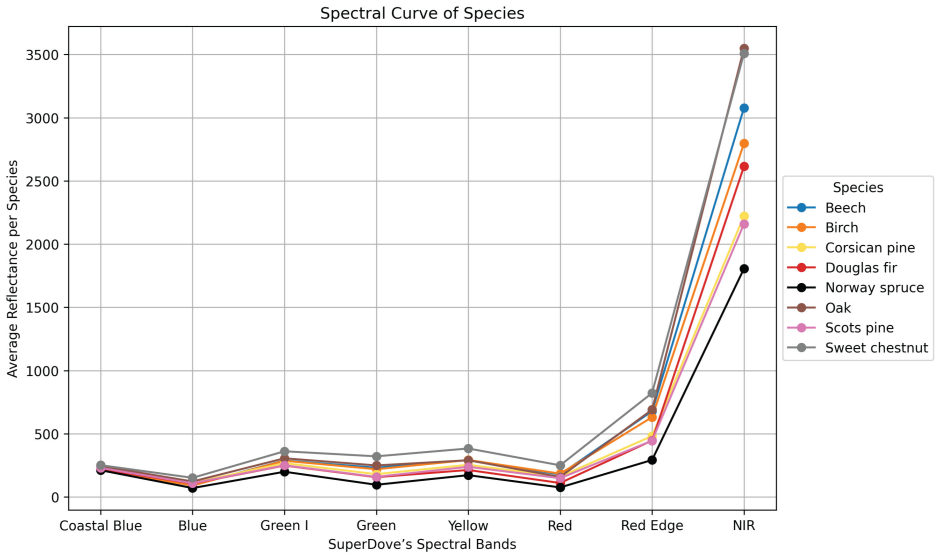

Figure 5.3: Spectral curves for different tree species across SuperDove's spectral bands.

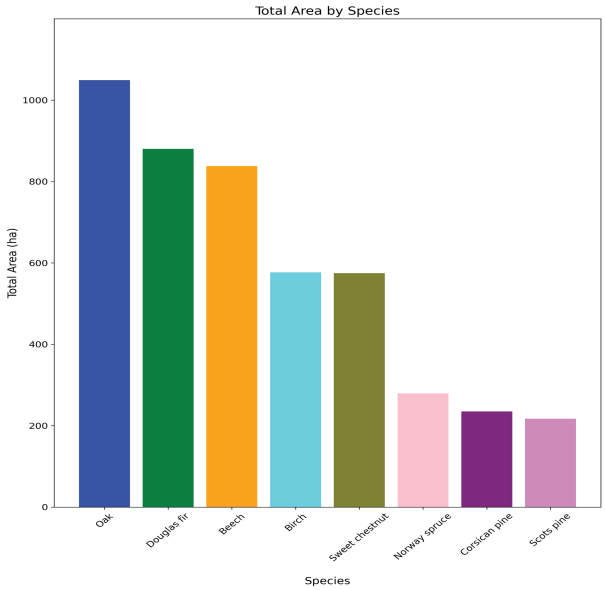

Figure 5.4: Comparison of the total area (in hectares) occupied by different tree species in the Forest of Dean.

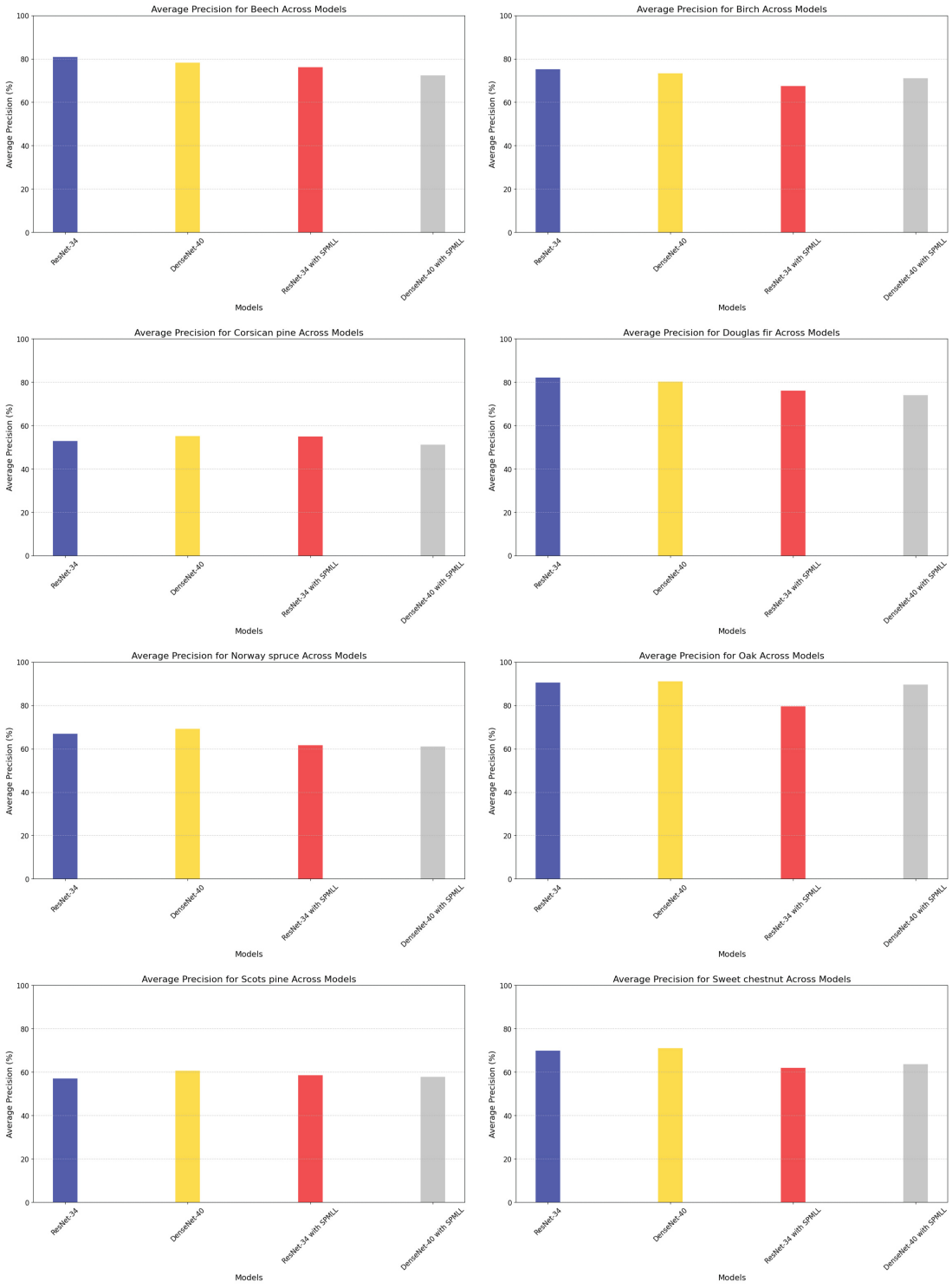

Figure 5.5: Average Precision (AP) scores for different tree species across various models. ”ResNet-34” and ”DenseNet-40” refer to models trained with fully labeled dataset, while ”ResNet-34 with SPMLL” and ”DenseNet-40 with SPMLL” refer to models trained with Single Positive Multi-Label Learning. Each subfigure represents the AP for a specific tree species, highlighting each model’s prediction performance.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this project, we classified different tree species in the Forest of Dean in a multi-label setting using high-resolution satellite imagery, employing ResNet-34 and DenseNet-40 CNN models. The study involved training models on both fully labeled Sub-Compartment Database (SCDB) and then using the Single Positive Multi-Label Learning (SPMLL) approach.

For the fully labeled SCDB, a comparative analysis of ResNet-34 and DenseNet-40 revealed their respective strengths and weaknesses. While the micro scores showed ResNet-34 performing slightly better than DenseNet-40, the mAP scores were almost identical, with DenseNet-40 showing a minutely higher mAP. However, this difference is negligible and may not be practically significant. The macro scores, on the other hand, showed more noticeable differences. To understand these differences, it was important to consider the distribution of tree species within the dataset. DenseNet-40 demonstrated higher AP scores for less frequent species, indicating its effectiveness at identifying true positives across varying thresholds. However, its lower macro average scores suggested that at a specific threshold, it might not identify as many true positives as ResNet-34 for these less frequent species. This discrepancy arises because AP evaluates the model's performance across various thresholds and takes a weighted average, providing a comprehensive measure, whereas macro metrics are calculated at a fixed threshold and may reflect better performance at that specific point. Also, it is important to note that while DenseNet-40 is more effective at identifying less frequent species across various thresholds, ResNet-34 achieves better results for abundant species within the same range of thresholds resulting in nearly identical mAP scores for both

models.

In analyzing the classification accuracy for various tree species, it was found that Oak had the highest classification metrics. Conversely, Corsican pine and Scots pine had the lowest metrics, reflecting the challenges in distinguishing these species due to their similar spectral features. Norway spruce, despite having a similar area coverage to Corsican pine and Scots pine, achieved relatively better classification metrics, benefiting from its distinctive spectral signatures compared to all other species.

When using the SPMLL approach, both models achieved performance metrics that were competitive to those obtained with fully labeled data. The ResNet-34 and DenseNet-40 models trained with SPMLL demonstrated the ability to leverage limited supervision effectively, maintaining comparable results to their full label supervision counterparts while significantly reducing the labeling effort for the SCDB. This ability to deliver strong performance with substantially less annotated data underscores the potential of SPMLL to alleviate the annotation burden. This reduction in labeling effort can translate into significant time and cost savings, making SPMLL a viable and cost-effective alternative in scenarios where extensive labeling is impractical or resource-intensive. Additionally, the successful application of SPMLL for the SCDB in this project demonstrates its scalability and applicability for further forestry applications.

Overall, the findings of this study demonstrate the robustness and practical utility of both CNN models and the SPMLL approach in multi-label classification of tree species. The ability to achieve high classification performance with minimal supervision makes the SPMLL method a valuable tools for forest management.

6.2 Future Work

In this project, the Single Positive Multi-Label Learning (SPMLL) models were trained using the weak assume negative (WAN) loss function. Future work could explore training SPMLL models with more advanced loss functions such as Regularized Online Label Estimation (ROLE) [5] and Expected Positive Regularization (EPR) [5]. These advanced loss functions have the potential to further enhance the performance of SPMLL models.

Also, incorporating satellite images from seasons such as spring, and autumn, could significantly benefit future research. By leveraging data from these seasons, models could be trained to recognize tree species under varying conditions, thereby improving their overall robustness and accuracy. Additionally, combining multi-seasonal data with

advanced loss functions could further enhance the capability of SPMLL models, making them more adaptable to real-world applications in forest management and ecological studies. This comprehensive approach would not only reduce annotation efforts but also provide a more detailed and accurate mapping of tree species.

Bibliography

[1] Gordon B. Bonan. Forests and climate change: Forcings, feedbacks, and the climate benefits of forests. *Science*, 320(5882):1444–1449, 2008.

[2] Emre Cakir, Toni Heittola, Heikki Huttunen, and Tuomas Virtanen. Polyphonic sound event detection using multi label deep neural networks. In *2015 International Joint Conference on Neural Networks, IJCNN 2015, Killarney, Ireland, July 12-17, 2015*, pages 1–7. IEEE, 2015.

[3] Di Chen, Yexiang Xue, Daniel Fink, Shuo Chen, and Carla P. Gomes. Deep multi-species embedding. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 3639–3646. ijcai.org, 2017.

[4] Matthew L. Clark and Dar A. Roberts. Species-level differences in hyperspectral metrics among tropical rainforest trees as determined by a tree-based classifier. *Remote. Sens.*, 4(6):1820–1855, 2012.

[5] Elijah Cole, Oisin Mac Aodha, Titouan Lorieul, Pietro Perona, Dan Morris, and Nebojsa Jojic. Multi-label learning from single positive labels. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 933–942. Computer Vision Foundation / IEEE, 2021.

[6] C. Dechesne, C. Mallet, A. Le Bris, V. Gouet, and A. Hervieu. Forest stand segmentation using airborne lidar data and very high resolution multispectral imagery. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B3:207–214, 2016.

[7] Jia Deng, Olga Russakovsky, Jonathan Krause, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Scalable multi-label annotation. In *CHI Conference on Human Factors in Computing Systems, CHI'14, Toronto, ON, Canada - April 26 - May 01, 2014*, pages 3099–3102. ACM, 2014.

[8] David Ellison, Martyn N. Futter, and Kevin Bishop. On the forest cover–water yield debate: from demand- to supply-side thinking. *Global Change Biology*, 18(3):806–820, 2012.

[9] Forestry England. The forest of dean. `https://www.forestryengland.uk/the-forest-of-dean`. Accessed: 14 July 2024.

[10] Rafael García, Juan C. Suárez, and Genevieve Patenaude. *Delineation of individual tree crowns for LiDAR tree and stand parameter estimation in Scottish woodlands*, pages 55–85. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[11] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pages 249–256. JMLR.org, 2010.

[12] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, volume 15 of *JMLR Proceedings*, pages 315–323. JMLR.org, 2011.

[13] Haoyu Gong, Qian Sun, Chenrong Fang, Le Sun, and Ran Su. Treedetector: Using deep learning for the localization and reconstruction of urban trees from high-resolution remote sensing images. *Remote. Sens.*, 16(3):524, 2024.

[14] Ian J. Goodfellow, Yoshua Bengio, and Aaron C. Courville. *Deep Learning*. Adaptive computation and machine learning. MIT Press, 2016.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016.

[16] Johannes Heinzel and Barbara Koch. Investigating multiple data sources for tree species classification in temperate forest and use for single tree delineation. *Int. J. Appl. Earth Obs. Geoinformation*, 18:101–110, 2012.

[17] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2261–2269. IEEE Computer Society, 2017.

[18] Markus Immitzer, Clement Atzberger, and Tatjana Koukal. Tree species classification with random forest using very high spatial resolution 8-band worldview-2 satellite data. *Remote. Sens.*, 4(9):2661–2693, 2012.

[19] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 448–456. JMLR.org, 2015.

[20] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2017–2025, 2015.

[21] Rie Johnson and Tong Zhang. Effective use of word order for text categorization with convolutional neural networks. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 103–112. The Association for Computational Linguistics, 2015.

[22] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomás Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, pages 427–431. Association for Computational Linguistics, 2017.

[23] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing*

Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States, pages 1106–1114, 2012.

[25] Planet Labs. Planetscope. `https://developers.planet.com/docs/data/planetscope/`. Accessed: 15 July 2024.

[26] Planet Labs. Understanding planetscope instruments. `https://developers.planet.com/docs/apis/data/sensors/`. Accessed: 15 July 2024.

[27] Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. Deep learning. *Nat.*, 521(7553):436–444, 2015.

[28] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

[29] Weijia Li, Runmin Dong, Haohuan Fu, and Le Yu. Large-scale oil palm tree detection from high-resolution satellite images using two-stage convolutional neural networks. *Remote. Sens.*, 11(1):11, 2019.

[30] Jiazheng Liu, Xuefeng Wang, and Tian Wang. Classification of tree species and stock volume estimation in ground forest images using deep learning. *Computers and Electronics in Agriculture*, 166:105012, 2019.

[31] Luxia Liu, Nicholas C. Coops, Neal W. Aven, and Yong Pang. Mapping urban tree species using integrated airborne hyperspectral and lidar remote sensing data. *Remote Sensing of Environment*, 200:170–182, 2017.

[32] Weiwei Liu, Haobo Wang, Xiaobo Shen, and Ivor W. Tsang. The emerging trends of multi-label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7955–7974, 2022.

[33] Hung-Yi Lo, Ju-Chiang Wang, Hsin-Min Wang, and Shou-De Lin. Cost-sensitive multi-label learning for audio tag annotation and retrieval. *IEEE Trans. Multim.*, 13(3):518–529, 2011.

[34] Lei Ma, Yu Liu, Xueliang Zhang, Yuanxin Ye, Gaofei Yin, and Brian Alan Johnson. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 152:166–177, 2019.

[35] Akira S. Mori, Kenneth P. Lertzman, and Lena Gustafsson. Biodiversity and ecosystem services in forest ecosystems: a research agenda for applied forest ecology. *Journal of Applied Ecology*, 54(1):12–27, 2017.

[36] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 807–814. Omnipress, 2010.

[37] Marion Pause, Christian Schweitzer, Michael Rosenthal, Vanessa Keuck, Jan Bumberger, Peter Dietrich, Marco Heurich, András Jung, and Angela Lausch. *In Situ*/remote sensing integration to assess forest health - A review. *Remote. Sens.*, 8(6):471, 2016.

[38] Michal Petr, Genevieve Patenaude, and Juan Suárez. *Forest Stand Volume of Sitka Spruce Plantations in Britain: Can Existing Laser Scanning Methods Based on the Conventional One Provide Better Results, a Comparison of Two Approaches*, pages 1–26. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[39] Steven J. Phillips, Miroslav Dudík, and Robert E. Schapire. A maximum entropy approach to species distribution modeling. In *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, volume 69 of *ACM International Conference Proceeding Series*. ACM, 2004.

[40] Alin-Ionut Plesoianu, Mihai-Sorin Stupariu, Ionut Cosmin Sandric, Ileana Patru-Stupariu, and Lucian Dragut. Individual tree-crown detection and species classification in very high-resolution remote sensing imagery using a deep learning ensemble model. *Remote. Sens.*, 12(15):2426, 2020.

[41] Yashoteja Prabhu and Manik Varma. Fastxml: a fast, accurate and stable tree-classifier for extreme multi-label learning. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 263–272. ACM, 2014.

[42] Ruiliang Pu and Shawn Landry. A comparative analysis of high spatial resolution ikonos and worldview-2 imagery for mapping urban tree species. *Remote Sensing of Environment*, 124:516–533, 2012.

[43] Ruiliang Pu and Shawn Landry. Mapping urban tree species by integrating multi-seasonal high resolution pléiades satellite imagery with airborne lidar data. *Urban Forestry & Urban Greening*, 53:126675, 2020.

[44] Timothy N. Rubin, America Chambers, Padhraic Smyth, and Mark Steyvers. Statistical topic models for multi-label document classification. *CoRR*, abs/1107.2462, 2011.

[45] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, October 1986.

[46] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014.

[47] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[48] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*, 2015.

[49] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *CoRR*, abs/1505.00387, 2015.

[50] Krzysztof Stereńczak, Bartłomiej Kraszewski, Miłosz Mielcarek, Żaneta Piasecka, Maciej Lisiewicz, and Marco Heurich. Mapping individual trees with airborne laser scanning data in an european lowland forest using a self-calibration algorithm. *International Journal of Applied Earth Observation and Geoinformation*, 93:102191, 2020.

[51] Fabien Hubert Wagner, Matheus Pinheiro Ferreira, Alber Sanchez, Mayumi C.M. Hirye, Maciel Zortea, Emanuel Gloor, Oliver L. Phillips, Carlos Roberto de Souza Filho, Yosio Edemir Shimabukuro, and Luiz E.O.C. Aragão. Individual tree crown delineation in a highly diverse tropical forest using very high resolution satellite

images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145:362–377, 2018. SI: Latin America Issue.

[52] Haoming Wan, Yunwei Tang, Linhai Jing, Hui Li, Fang Qiu, and Wenjin Wu. Tree species classification of forest stands using multisource remote sensing data. *Remote. Sens.*, 13(1):144, 2021.

[53] Jeremy M. Wolfe. Visual search. *Current biology : CB*, 20(8):R346–349, April 2010. Place: England.

[54] Jeremy M. Wolfe, Todd S. Horowitz, and Naomi M. Kenner. Cognitive psychology: rare items often missed in visual searches. *Nature*, 435(7041):439–440, May 2005. Place: England.

[55] Bin Wu, Erheng Zhong, Andrew Horner, and Qiang Yang. Music emotion recognition by multi-label multi-layer multi-instance multi-view learning. In *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*, pages 117–126. ACM, 2014.

[56] Donna Xu, Yaxin Shi, Ivor W. Tsang, Yew-Soon Ong, Chen Gong, and Xiaobo Shen. Survey on multi-output learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31(7):2409–2429, 2020.

[57] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I*, volume 8689 of *Lecture Notes in Computer Science*, pages 818–833. Springer, 2014.

[58] Min-Ling Zhang and Zhi-Hua Zhou. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2014.

Appendix A

Additional Figures

A.1 Tree Species Distribution in the Forest of Dean

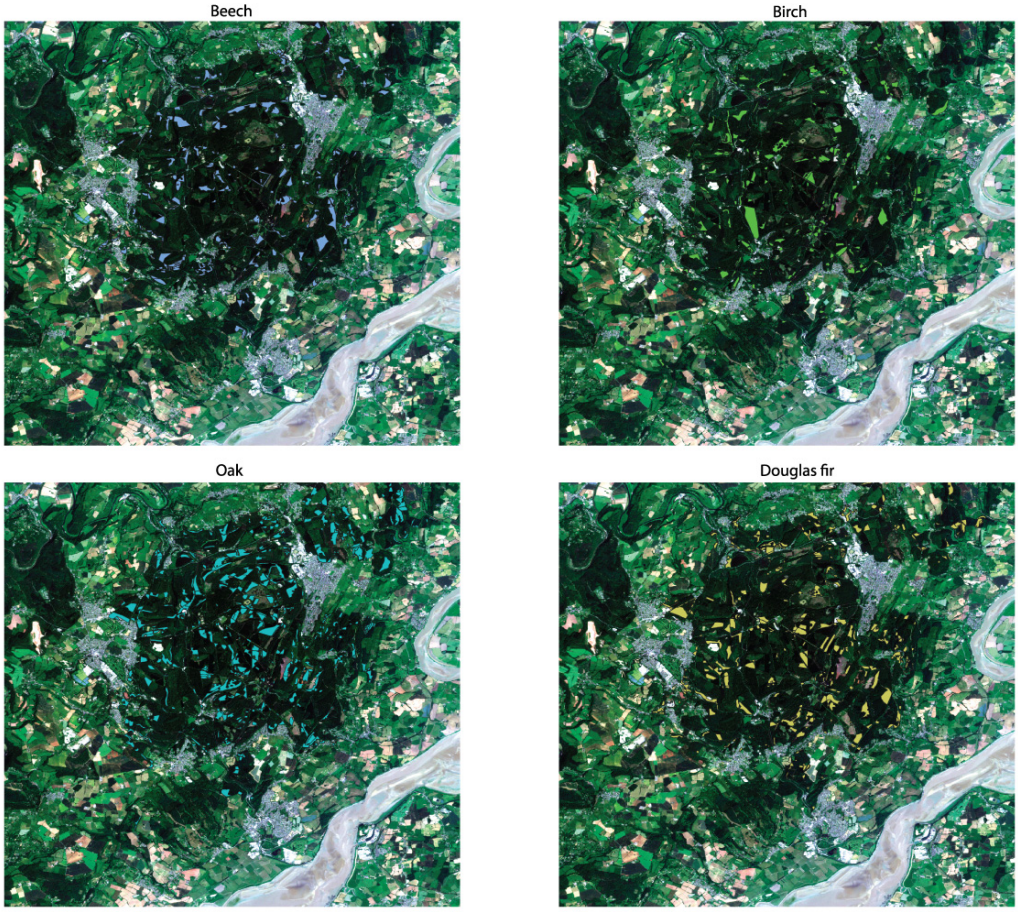

Figure A.1: Spatial distribution of Beech, Birch, Oak, and Douglas fir across the area of interest (AOI), highlighting the varying density of each species. (Figure created for this project using QGIS software.)

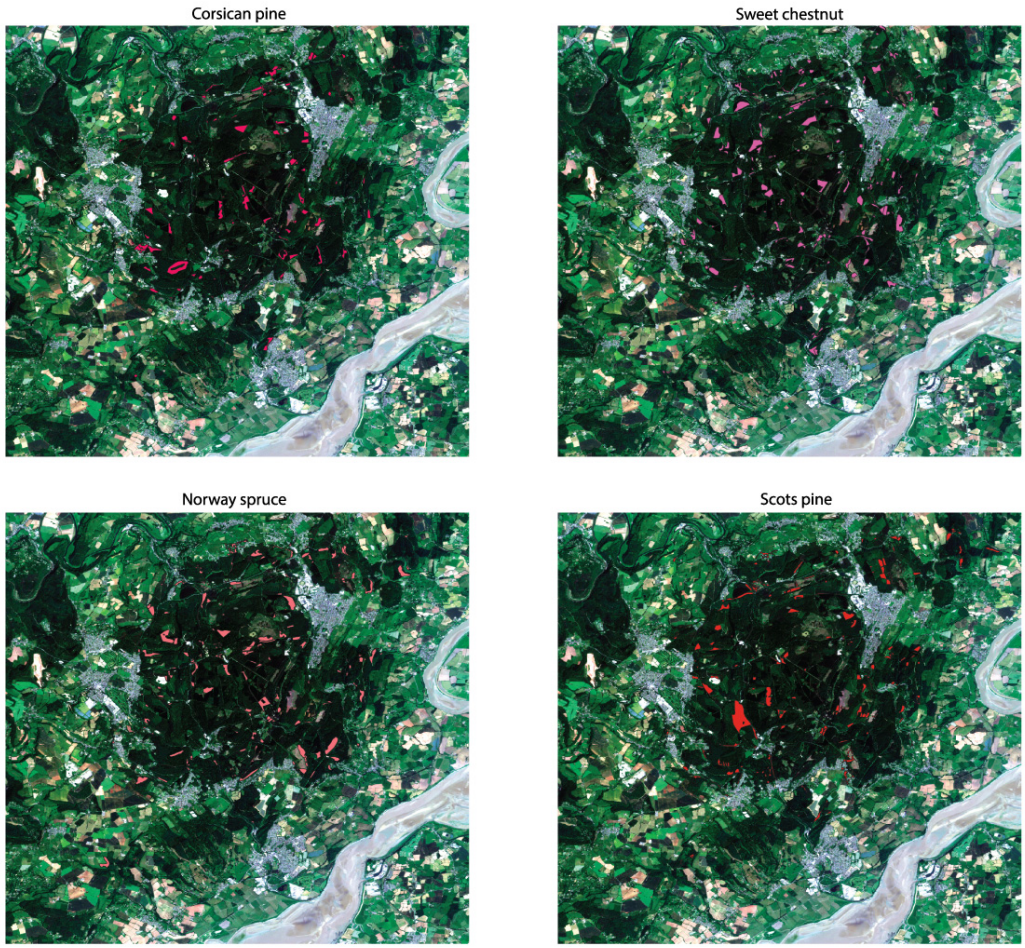

Figure A.2: Spatial distribution of Corsican pine, Sweet chestnut, Norway spruce, and Scots pine trees across the area of interest (AOI), highlighting the varying density of each species. (Figure created for this project using QGIS software.)

Figure A.1 illustrates the distribution patterns of the more abundant tree species, emphasizing their prevalence across the area of interest. In contrast, Figure A.2 highlights the distribution of species that are less abundant, showcasing their more limited presence.