

Thompson Rivers University

“Multi-scale spatial image analysis for mapping spotted knapweed  
in grassland ecosystems”

by

Shohreh Sahebi

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE OF  
Master of Science in Environmental Science

KAMLOOPS, BRITISH COLUMBIA

March 2024

Dr. David Hill

Dr. Wendy Gardner

Dr. Musfiq Rahman

This study investigates the value of scale-space representations of spatial features for the identification and mapping of invasive plant species in high-resolution multi-spectral imagery. Scale-space representations combine the spatial domain with a scale dimension such that spatial features can be represented at multiple-spatial scales. In this work, Gaussian pyramids (GPs) are constructed to create discrete representations of the spatial features across the scale-space. A case study is employed to evaluate the performance of classifiers constructed using features at various levels within the GP scale-space to a classifier constructed using all features across the scale space. This case study explores the identification and mapping of spotted knapweed in a grassland ecosystem using multispectral imagery acquired using a remotely piloted aircraft system (RPAS). Given the large number of features, feature optimization was critical for developing high-performing classifiers. Classification was performed using two machine learning classifiers, random forest and support vector machine (SVM). The results of this case study show that very high-spatial resolution features do not produce the best image classifications, but rather that there is an optimal scale, lower than that of the raw imagery, of the image features that produces the best classification accuracy. The results also show that classification is not improved by the inclusion of features at multiple spatial scales. These findings suggest not only that feature spatial scale optimization can improve image analysis, but also that this optimization can inform RPAS flight planning to improve mission efficiency.

Keywords: Remote sensing, Gaussian pyramid, Scale-Space Random forest, Support vector machine, Invasive plants, *Centaurea stoebe*, Remotely piloted aircraft systems (RPAS)

## Table of Contents

<b>Chapter 1 Introduction</b> .....	<b>8</b>
<b>Introduction</b> .....	<b>9</b>
<b>Preserving Grassland Ecosystems</b> .....	<b>9</b>
<b>Remote Sensing Techniques for Mapping Invasive Plant Species</b> .....	<b>10</b>
<b>Enhancing Plant Species Invasion Monitoring with Remotely Piloted Aircraft Systems</b> .....	<b>11</b>
<b>Machine Learning Algorithms for Mapping Invasive Plants</b> .....	<b>12</b>
<b>Objectives</b> .....	<b>13</b>
<b>Study Site</b> .....	<b>14</b>
<b>Data Acquisition and Image Processing</b> .....	<b>15</b>
<b>Roadmap of the Thesis</b> .....	<b>19</b>
<b>Chapter 2 Method</b> .....	<b>21</b>
<b>Introduction</b> .....	<b>22</b>
<b>Machine Learning: an Overview</b> .....	<b>23</b>
Data Partitioning: Training and Validation Data.....	26
Cross Validation.....	27
Recursive Feature Elimination (RFE) .....	29
<b>Random forest</b> .....	<b>30</b>
Optimizing RF performance: Hyperparameters Tuning .....	32
<b>Support Vector Machine (SVM)</b> .....	<b>34</b>
Optimizing SVM Performance: Hyperparameters Tuning.....	37
<b>Optimizing Machine Learning Performance: Grid Search</b> .....	<b>38</b>
<b>Gray Level Co-occurrence Matrix (GLCM) Based Texture Analysis</b> .....	<b>38</b>
Mathematical Representation of GLCM-based Features.....	41
<b>Introduction of Scale-Space Theory and the Gaussian Pyramid</b> .....	<b>43</b>
Importance of Multi-scale Analysis .....	44

Axiomatic Derivations and the Gaussian Approach.....	44
Gaussian Convolution and Scale-Space .....	45
Gaussian Pyramids .....	47
<b>Conclusion .....</b>	<b>50</b>
<b><i>Chapter 3 Invasive Species Mapping: A Case Study on Spotted Knapweed</i></b>	
<b><i>Detection in Grassland Ecosystems.....</i></b>	<b><i>52</i></b>
<b>Introduction.....</b>	<b>53</b>
<b>Methods.....</b>	<b>53</b>
Feature Extraction.....	54
Feature Compilation .....	57
Model Creation .....	58
<b>Results.....</b>	<b>60</b>
Feature Selection .....	60
Hyperparameter Optimization.....	68
Classifier Performance .....	70
<b>Discussion.....</b>	<b>87</b>
Mapping Spotted Knapweed at Site 3 .....	91
<b>Conclusion .....</b>	<b>97</b>
<b><i>Chapter 4 Conclusion &amp; Future works.....</i></b>	
<b>Conclusion .....</b>	<b>100</b>
<b>Future Work - Expanding Methodologies .....</b>	<b>102</b>
Generative Adversarial Networks for Training Data Augmentation .....	102
Feature Optimization through Data Compression.....	103
Other suggestions .....	104
<b><i>REFERENCES .....</i></b>	<b><i>106</i></b>

**List of Figures:**

*Figure 1: Locations of the 3 field sites in Laurie Guichon Memorial Grasslands Interpretive Site (LGMGIS) explored in this study. This map uses the WGS 84 UTM Zone 10N coordinate system. 14*

*Figure 2: A diagram demonstrating threefold cross-validation. Symbols represent training set samples, divided into three groups. Sequentially, each group is excluded during model training. Performance estimates, such as error rate, are derived for each withheld sample set. Averaging these performance metrics gives the cross-validation estimate of model performance. ....28*

*Figure 3: Decision tree structure showcasing root nodes, decision nodes, leaf nodes, and branches.....31*

*Figure 4: Illustration of an SVM classifier with a linear kernel applied to linearly separable data, highlighting the optimal separating hyperplane (solid line) and the margins (dashed lines) defined by the support vectors (circled points).....35*

*Figure 5: Visualization of a non-linear SVM classification, showing how a non-linear boundary effectively separates two distinct classes in the feature space. ....36*

*Figure 6: Illustration of the kernel trick applied to data in a 3D space, demonstrating the transformation that facilitates the separation of classes which are not linearly separable in the original dimensions.....36*

*Figure 7: Illustration of the four primary directions (0°, 45°, 90°, and 135°) used in the computation of Haralick texture features for GLCM analysis, depicting the relative positioning of pixel pairs.....39*

*Figure 8: 3D representation of a 2D uniform Gaussian kernel, visualizing the symmetrical distribution and peak concentration at the origin.....46*

*Figure 9: Gaussian Pyramid. The image illustrates five levels of the GP, spanning from the original image at level I0 to the fifth level, I4.....48*

*Figure 10: Gaussian Pyramid. The image illustrates five levels of the GP, spanning from the original image at level I0 to the fifth level, I4. Note the extent of the physical domain represented by each image stays the same despite the reduction in the number of number of pixels constituting the image.....55*

*Figure 11: A procedure of classifying Spotted Knapweed using RF and SVM classifiers. ....60*

*Figure 12: RFECV Results for GPO, showing the optimal selection of 21 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations. ....61*

<i>Figure 13: RFECV Results for GP1, showing the optimal selection of 16 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations. ....</i>	<i>62</i>
<i>Figure 14: RFECV Results for GP2, showing the optimal selection of 4 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations. ....</i>	<i>63</i>
<i>Figure 15: RFECV Results for GP3, showing the optimal selection of 16 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations. ....</i>	<i>64</i>
<i>Figure 16: RFECV Results for GP4, showing the optimal selection of 8 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations. ....</i>	<i>65</i>
<i>Figure 17: RFECV Results for GPs, showing the optimal selection of 5 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations. ....</i>	<i>66</i>
<i>Figure 18: RF confusion matrix for the GP0 validation set. ....</i>	<i>71</i>
<i>Figure 19: SVM confusion matrix for GP0 validation set. ....</i>	<i>71</i>
<i>Figure 20: RF confusion matrix for the GP1 validation set. ....</i>	<i>74</i>
<i>Figure 21: SVM confusion matrix for the GP1 validation set. ....</i>	<i>74</i>
<i>Figure 22: RF confusion matrix for the GP2 validation set. ....</i>	<i>76</i>
<i>Figure 23: SVM confusion matrix for the GP2 validation set. ....</i>	<i>76</i>
<i>Figure 24: RF confusion matrix for the GP3 validation set. ....</i>	<i>79</i>
<i>Figure 25: SVM confusion matrix for the GP3 validation set. ....</i>	<i>79</i>
<i>Figure 26: RF confusion matrix for the GP4 validation set. ....</i>	<i>82</i>
<i>Figure 27: SVM confusion matrix for the GP4 validation set. ....</i>	<i>82</i>
<i>Figure 28: RF confusion matrix for the concatenated GPs validation set. ....</i>	<i>85</i>
<i>Figure 29: SVM confusion matrix for the concatenated GPs validation set. ....</i>	<i>85</i>
<i>Figure 30: VNIR generated image from flight data collected at field site 3 on July 4, 2018. ....</i>	<i>94</i>
<i>Figure 31: RF Classification map generated using GLCM-GP0 meta pixel-based image analysis, illustrating the relative abundance of spotted knapweed. ....</i>	<i>95</i>

*Figure 32: RF Classification map generated using GLCM-GP2 meta pixel-based image analysis, illustrating the relative abundance of spotted knapweed. This level demonstrates the highest accuracy. ....96*

**List of Tables:**

*Table 1: Band number, band names, central wavelength, and full width at half maximum (FWHM) of Parrot Sequoia sensor. ....15*

*Table 2: Parameters adopted during flight data collection. ....16*

*Table 3: Relation between each level of GP and GSD. ....55*

*Table 4: GLCM Extracted Features. ....56*

*Table 5: Classification of Spotted Knapweed Abundance in Surveyed Sites. ....57*

*Table 6: Classifier performance metrics used to evaluate classifiers performance. ....59*

*Table 7: Optimized Features for GP0, GP1, GP2, GP3, GP4 and concatenated GPs. ....67*

*Table 8: Range of hyperparameter values considered for tuning the Random Forest Classifier using Grid Search. ....68*

*Table 9: Range of hyperparameter values considered for tuning SVM using Grid Search. ....68*

*Table 10: Result of RF hyperparameters tuning for GP0 to GP4 and GPs. ....69*

*Table 11: Result of SVM hyperparameters tuning for GP0 to GP4 and GPs. ....70*

*Table 12: RF and SVM classification results for the GP0 feature set. ....72*

*Table 13: RF and SVM classification results for the GP1 feature set. ....75*

*Table 14: RF and SVM classification results for the GP2 feature set. ....77*

*Table 15: RF and SVM classification results for the GP3 feature set. ....80*

*Table 16: RF and SVM classification results for the GP4 feature set. ....83*

*Table 17: RF and SVM classification result for the concatenated GPs feature set.....86*

# Chapter 1

## Introduction



## Introduction

Invasive plants are non-native species that have become established in new environments, impacting the structure and function of the existing ecosystems and out-competing native biotic communities (Pyšek and Richardson, 2010; Qian and Ricklefs, 2006). The estimated economic loss caused by all types of invasive species has been at least \$1.288 trillion (U.S. dollars) worldwide since 1970 (Diagne et al., 2021). One notable example in the grasslands and woodlands of western North America is *Centaurea stoebe*, or spotted knapweed. This species has become a significant management challenge as it has spread across millions of hectares, resulting in significant financial repercussions both from control measures and decreased forage yield (Singh et al., 2022).

Land managers recognize the importance of intensive monitoring and early detection in effectively managing invasive species (Hobbs and Humphries, 1995). However, controlling invasions can be challenging due to the large size and complexity of invaded ecosystems (Holden, Nyrop, and Ellner, 2016). Early detection of invasive species has been shown to enhance the cost-effectiveness of treatment strategies (Malanson and Walsh, 2013; Holden et al., 2016). For this reason, accurate and reliable methods for early detection of invasive species are vital. These methods often involve a combination of surveillance, monitoring, and rapid response systems. Advanced technologies, such as remote sensing (RS), DNA analysis, and predictive modeling, are increasingly being employed to improve early detection capabilities (Cassidy, 2020; Martinez et al., 2020).

## Preserving Grassland Ecosystems

Detecting and mapping species invasions in grasslands is essential for managing these sensitive ecosystems. Grasslands, including both sown pasture and rangeland, comprise some of the largest ecosystems worldwide, representing approximately 20 to 40 percent of the Earth's land area (Suttie et al., 2005). Natural grasslands are one of the most endangered ecosystems in North America (Samson and Knopf, 1994). Grasslands provide irreplaceable ecosystem services to people and the environment (O'Mara, 2012). However, human utilization of these grasslands can inadvertently promote the spread of

invasive plants, leading to the displacement of native species and a decline in land values, as well as ecological goods and services (Foster et al., 2020; Gaskin et al., 2021).

## Remote Sensing Techniques for Mapping Invasive Plant Species

Field-based visual inspections and plant species inventories are often used for mapping plant species invasions; however, these methods are time-consuming and impractical for large areas (Bradley, 2014). To address these limitations, the potential of RS technology, known for its ability to collect data over vast spatial extents, has been extensively explored for mapping species invasions (Bradley, 2014; Huang and Asner, 2009; Joshi et al., 2004). Remote sensing techniques have shown promise in mapping invasive plants based on various plant characteristics such as seasonal phenology, biochemical, physiological, and structural characteristics, as well as the prevalence of invasive species in the study area (Gholizadeh et al., 2022).

Many studies have investigated multispectral and hyperspectral imagery to identify invasive plants during flowering and fruiting when these plants exhibit a distinct spectral response from surrounding green plants (Andrew and Ustin, 2009; Huang and Geiger, 2008; Ishii and Washitani, 2013). However, the success of these approaches depends not only on the distinct phenology of target invasive plants but also on access to RS data collected when the target plants are in a specific phenological stage, requiring either time-targeted image acquisition or high-temporal resolution imagery throughout the growing season. High spatial resolution hyperspectral imagery has been used to detect invasive plants based on how their biochemical, physiological, and/or structural traits affect their spectral response (Glenn et al., 2005; Mitchell and Glenn, 2009; Yang and Everitt, 2010), due to the costs of acquiring high-resolution hyperspectral imagery, however, this approach is less common.

The primary challenge in vegetation mapping using RS is precisely differentiating target plants from background vegetation. This challenge is even more pronounced in grasslands due to the small size and sparse canopy of vegetation (Malanson and Walsh, 2013). Effective detection often necessitates that the invasive plants form patches reasonably uniform in nature and larger than the spatial resolution of the RS imagery (He

et al., 2015). Such observations underscore the significance of image spatial resolution in the mapping process (Underwood et al., 2007).

Satellite-acquired multispectral imagery offers spatial resolutions that range from tens of meters to meters. Several studies have explored the use of imagery acquired by the NASA/USGS Landsat-8 program (Khare et al., 2018; Matongera et al., 2017; Royimani et al., 2019) and the ESA Sentinel-2 constellation (Duncan et al., 2023; Gholizadeh et al., 2022; Hawryło et al., 2018; Rupasinghe and Chow-Fraser, 2021) for mapping invasive species. However, with resolutions of 30 m and 10-20 m, respectively, imagery from these satellite programs are ill-suited for mapping invasive plants with small or sparse canopies, particularly in the early stages of invasion (Malanson and Walsh, 2013).

For detecting small or fragmented patches of invasive species, researchers have explored the privately operated World View Satellite program and the ESA Planet Scope program acquire imagery. Imagery from these systems have spatial resolutions of 1.84m and 3m, respectively (Lake et al., 2022; Shiferaw et al., 2019). However, even meter-scale imagery is too coarse to support the mapping of map invasives species with very small or very sparse canopies (Malanson and Walsh, 2013).

## Enhancing Plant Species Invasion Monitoring with Remotely Piloted Aircraft Systems

Recently, there has been increasing interest in using remotely piloted aircraft systems (RPASs) as an RS platform for monitoring and mapping plant species invasions (Dvořák et al., 2015; Hill et al., 2020; Lehmann et al., 2017; Mallmann et al., 2020). One of the key advantages of using RPAS is their ability to capture spectral images with unprecedented levels of spatial and spectral resolutions (Hill et al., 2020); this means that the acquired data can provide highly detailed and accurate measurements of the target vegetation. Another significant benefit of RPASs is the ease and flexibility in deploying these systems for imaging missions. Unlike traditional aerial or satellite platforms, RPASs can be quickly launched and maneuvered over specific areas of interest; this enables researchers and land managers to conduct imaging missions with high frequency,

increasing the temporal resolution of the collected data (Hill et al., 2020; Klosterman et al., 2018; Klosterman & Richardson, 2017).

Access to high-resolution imagery provided by RPASs has revolutionized the field of invasive species monitoring and management. High resolution RPAS-acquired imagery allows for detecting and measuring target species within treatment areas with remarkable precision (Martin et al., 2018; Hill et al., 2017; Tamouridou et al., 2017). Researchers can identify and quantify the extent of invasive species presence, monitor their spread, and assess the effectiveness of control measures. This level of detail and accuracy is particularly valuable when dealing with very small or very sparse canopies, where satellite-based imagery may not be as effective (Gholizadeh et al., 2022; Malanson & Walsh, 2013).

## Machine Learning Algorithms for Mapping Invasive Plants

Machine learning algorithms (MLAs) have become a powerful tool in RS image analysis, due to their ability to model highly dimensional and non-linear data with complex interactions and overcome challenges associated with data gaps (Thessen, 2016). While many machine learning-based classification techniques exist, both parametric and non-parametric, their applications extend beyond just RS classifications. For instance, these techniques have found utility in areas such as medical imaging (Erickson et al., 2017), financial forecasting (Kamalov et al., 2021), and text classification (Ikonomakis et al., 2005). Within the realm of RS, they play a pivotal role in tasks like land cover mapping (Petropoulos et al., 2012), vegetation health monitoring (Hawryło et al., 2018; Selvaraj et al., 2020), and urban development tracking (Shafizadeh-Moghadam et al., 2017). Machine learning algorithms can effectively learn and model these complex patterns, enabling the identification and classification of invasive plants with improved accuracy and precision compared to traditional manual methods or basic automated processes (Mountrakis et al., 2011). Furthermore, the integration of ancillary data, such as environmental variables or topographic features, can further enhance the classification performance (Ng et al., 2016; Nininahazwe et al., 2023).

## Objectives

In the southern interior of British Columbia, the invasive spotted knapweed (*Centaurea stoebe*) is not only a growing concern but has also been categorized under the Regional Containment/Control priority categories established by Provincial Priority Invasive Species BC (Inter-Ministry Invasive Species Working Group March 2021). Recently Baron and Hill (2020) developed a method, called metapixel-based image classification, for mapping spotted knapweed in a grassland ecosystem using RPAS-acquired multispectral imagery. The metapixel-based classification segments the study area into non-overlapping squares larger than the image resolution, termed metapixels, to derive spectral features from the image pixels within each metapixel. Baron and Hill (2020) used a metapixel size of 1m<sup>2</sup>, corresponding to the size of quadrats commonly used by range managers, to determine the relative abundance of target species. Their study showed that by applying this method and using second-order spatial statistics derived from the grey level co-occurrence matrix (GLCM) of the metapixels (Haralick et al., 1973), the relative abundance of spotted knapweed in each metapixel could be determined with an overall accuracy of 66.0% when validated with an independent dataset (Baron and Hill, 2020).

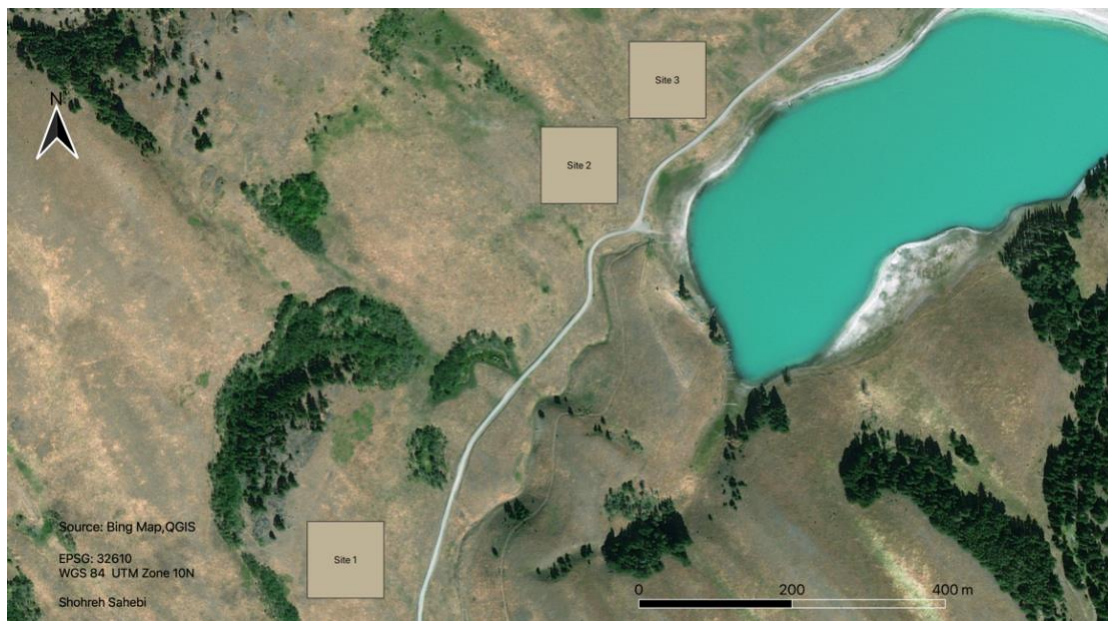
This study aims to expand on the previous work by exploring the impact of image spatial resolution on mapping spotted knapweed in a grassland ecosystem. Due to weight limitations, RPAS-based RS often employs less accurate sensors than satellite or conventional aircraft-based RS (Hill et al., 2019). Increasing the area corresponding to a pixel, called the ground-resolved distance, in an image increases the amount of spatial averaging in determining the measurement associated with that pixel. While this averaging can enhance the image smoothness by reducing high-frequency details, it also amplifies the within-pixel variability of the captured spectral data, leading to image blurring. Building on these insights, I hypothesize that there exists an optimal spatial resolution for spectral features used in classifying spotted knapweed within multispectral imagery. It's crucial to note that when referring to "features," the emphasis is on the descriptive attributes of the metapixel, rather than physical elements present within the image scene, such as trees or grass.

To investigate these hypotheses, this study is driven by two primary objectives:

- 1) Determine if there is a relationship between spatial resolution and image classification accuracy that can help identify an optimal spatial resolution for mapping spotted knapweed using multispectral imagery.
- 2) Determine if the image features at multiple spatial resolutions improve or hinder the identification of spotted knapweed using multispectral imagery.

## Study Site

The data that will be used in this work was collected within the Laurie Guichon Memorial Grasslands Interpretive Site (LGMGIS), which is located south of Merritt, British Columbia (BC), Canada. This 100-hectare site is situated in Canada's Western Cordillera physiographic region and is classified as representing BC's Interior Douglas Fir dry hot ecosystem subzone. In a previous study conducted by Baron and Hill (2020), three field sites were selected within the LGMGIS. Each field site covered an approximate area of 1 hectare and exhibited a gentle slope with a southern aspect. Figure 1 shows the locations of these three field sites within the LGMGIS.



*Figure 1: Locations of the 3 field sites in Laurie Guichon Memorial Grasslands Interpretive Site (LGMGIS) explored in this study. This map uses the WGS 84 UTM Zone 10N coordinate system.*

## Data Acquisition and Image Processing

The data used in this research was collected by Jackson Baron as part of his thesis work (Baron, 2019). These data were collected using a Parrot Sequoia multispectral sensor, featuring a 16-megapixel digital camera and four 1.2-megapixel global shutter single-band imagers, accompanied by an incident light sensor and GPS. The sensor was securely mounted on a DJI Phantom 4 RPAS for aerial data acquisition. Imaging flights were carried out between 11:00 a.m. and 1:00 p.m. July 04, July 12, and July 19, 2018. On July 04, imagery was acquired at all 3 field sites. However, due to technical difficulties with the Phantom 4 RPAS, data was not acquired from Site 1 on July 12 or from Site 3 on July 19.

The Parrot Sequoia sensor, which features a 16-megapixel camera and an in-built GPS, was the primary tool. The sensitivities of this sensor's single-band imagers are detailed in Table 1. For optimal results considering both safety and resolution, a flight altitude of 30m was maintained. All conducted flights were in compliance with the 2018 Canadian Aviation Regulations. More specifics on the flight parameters can be found in Table 2.

*Table 1: Band number, band names, central wavelength, and full width at half maximum (FWHM) of Parrot Sequoia sensor.*

Band Number	Nominal Reflectance	Centered Wavelength	FWHM (nm)
1	Green	550	40
2	Red	660	40
3	Red Edge	735	10
4	NIR	790	40

Table 2: Parameters adopted during flight data collection.

Height Above Ground Level(m)	Forward Overlap (%)	Side Overlap (%)	Time of Day	Max Wind Speed (km/h)
30	80	80	11:00a.m. - 1:00p.m.	40

Every scene captured by the Sequoia's single-band imagers assigns a digital number (DN) to each pixel. These DNs are linked to the radiance (measured in  $\text{Wm}^{-2} \text{sr}^{-1}$ ) reflected from the land surface over the pixel area. This relationship can be expressed through an equation provided by Parrot (2017).

$$L = f^2 \frac{DN-B}{A\varepsilon\gamma+C} \quad (1)$$

Where  $DN$  represents the digital value assigned to each pixel. The exposure time of the image, given in seconds, is denoted by  $\varepsilon$ . The ISO is represented by  $\gamma$ . The f-number of the imager, symbolized by  $f$ , is set at 2.2, and it provides the relationship between the focal length and the aperture diameter of the lens. Calibration coefficients are denoted by  $A$ ,  $B$ , and  $C$ . These specific values are stored within the exchangeable image file (EXIF) metadata during the image capture process.

Additionally, during image capture by the single-band imagers, the Sequoia sensor's incident light sensor records a radiance level, denoted as  $\Psi$ . This radiance is associated with the irradiance,  $E$  (measured in  $\text{Wm}^{-2} \text{sr}^{-1}$ ) that the land surface receives over the area represented by a pixel. This relationship is defined according to a specific equation, as cited from Tu et al. (2018).

$$E = a \frac{\Psi}{G\Gamma} \quad (2)$$

Where  $E$  is the irradiance on the land surface,  $a$  represents a constant,  $\Psi$  denotes the radiance detected by the sensor. Additionally,  $G$  signifies the sensor's gain, and  $\Gamma$  stands for the time taken for measurement acquisition. The data values for  $\Psi$ ,  $G$ , and  $\Gamma$  are documented in the image's EXIF metadata when the image is captured.



The surface reflectance, represented as  $\rho$ , is determined in the post-processing stage of the images taken from the single-band imagers, and this is based on a specific formula provided by Tu et al. (2018).

$$\rho = K \frac{L}{E} \quad (3)$$

Where  $L$  is computed using Equation 1,  $E$  is computed using Equation 2, and  $K$  is a normalization constant related to the ratio of the solid angles from the incident light sensor to that of each pixel within the imager.

For each flight, the normalization constant  $K$  is approximated using data from a ground-based, calibrated reflectance target. A Parrot Sequoia Calibration Target (Parrot, Paris, France) with known reflectance values (green: 18.4%, red: 19.7%, red edge: 22.7%, NIR: 27.6%) was positioned near the take-off and landing sites, and images of this target were captured at the beginning and end of each flight. During the processing phase, the Pix4Dmapper photogrammetry and RPAS mapping software, an implementation of the structure-from motion (SfM) algorithm (Turner et al., 2012) identified the calibration target in the imagery, assisting in the estimation of the normalization constant. Given the known reflectance of the calibration target, equations 1, 2, and 3 assist in estimating the normalization constant ( $K$ ) from the image pixels linked to the calibration target.

The RPAS images were processed and combined using the Pix4Dmapper software, which applies the structure-from-motion algorithm. The orthomosaicked images produced by Pix4D are georeferenced using coordinates from the global navigation satellite system (GNSS) embedded in the Sequoia sensor. However, this georeferencing can be regarded as a preliminary or 1st order approximation, as the GNSS positioning system can have errors, potentially up to 10 meters. To enhance accuracy, 6 Ground Control Points (GCPs) per site were employed. This processing created orthomosaicks from the four calibrated single-band sensors including Green, Red, Near Infrared (NIR), and Red-Edge with a ground-resolved distance of 2.9cm.

For field surveys, specific locations were located on the orthomosaicks to extract sub-images. These locations were identified visually and manually digitized based on markers. The GPS's 60cm accuracy meant visual marker identification was more reliable. Each sub-

image, equivalent to a 1m<sup>2</sup> metapixel, represented an area surveyed in the field, containing 1156 pixels (34 rows by 34 columns).

Given the variability in spotted knapweed density across the surveyed quadrats and the relatively small number of quadrats surveyed, percent cover of spotted knapweed within each quadrat was categorized into qualitative classes. Quadrats in which spotted knapweed was not present or only present in trace were classified as “None”, quadrats in which spotted knapweed did not exceed 25% cover were classified as “Moderate”, and quadrats in which spotted knapweed exceeded 25% cover were classified as “High”. This categorization was essential for effectively training a classifier because it increased the number of examples representing each class of spotted knapweed cover. The classification was based on ensuring a balance between showing distribution and accounting for less-represented spotted knapweed concentrations. The complete dataset consists of 181 measured quadrats, 51 classified as None, 63 classified as Moderate, and 67 classified as High.

Subsequently, a comprehensive set of 84 features was extracted for each metapixel. This includes:

- Eight features that captured both the mean and standard deviation of reflectance values at the pixel level for every spectral band.
- Six features derived from the calculation of the mean and standard deviation for three multiband spectral indices for each pixel within the metapixel.
- The remaining 70 features were obtained through GLCM-based texture analysis, yielding valuable texture features for each of the five spectral bands and three multiband indices.

For each pixel within the metapixels, three multiband spectral indices were computed. These were derived from the normalized difference vegetation index (NDVI) and calculated by contrasting Near Infrared (NIR) with the remaining spectral reflectance bands. Indices computed using the reflectance values from the red and NIR bands are designated as NDVI (Carlson and Ripley, 1997). Those comparing the green and NIR band reflectance values are denoted as gNDVI. Lastly, indices comparing the red-edge and NIR

band reflectance values are labeled as reNDVI. Each index was computed as follows (Baron and Hill, 2020):

$$NDVI = \frac{\rho_{NIR} - \rho_R}{\rho_{NIR} + \rho_R} \quad (4)$$

$$gNDVI = \frac{\rho_{NIR} - \rho_G}{\rho_{NIR} + \rho_G} \quad (5)$$

$$reNDVI = \frac{\rho_{NIR} - \rho_{RE}}{\rho_{NIR} + \rho_{RE}} \quad (6)$$

Where  $\rho_G$ ,  $\rho_R$ ,  $\rho_{RE}$ , and  $\rho_{NIR}$  represent the reflectance values of the green, red, red-edge, and NIR spectral bands, respectively.

## Roadmap of the Thesis

In conclusion, Chapter 1 provides an overview of the research problem, the motivation behind the study, and the objectives set to be achieved. Key concepts and theories that form the foundation of the research as well as an orientation to the dataset explored in this work are also introduced. Following chapters will explore the methodologies employed in the study, followed by the presentation of the results obtained.

Chapter 2 - Methodology and Experimental Setup: This chapter delves into the research techniques utilized to meet the objectives of this work.

Chapter 3 - Results and Discussion: This chapter reports and analyses the research outcomes. It covers the discoveries obtained from employing the methods mentioned in Chapter 2 on the chosen dataset. A thorough analysis of the results is presented, bolstered by visuals and performance metrics. The chapter also integrates these findings and their contribution into the existing body of knowledge regarding RS-based invasive species mapping.

Chapter 4 - Recommendations and Future Work: This chapter will encapsulate the recommendations stemming from the research's key findings. It will shed light on

actionable suggestions that can be implemented to address current challenges and gaps identified. Moreover, a forward-looking perspective will be provided, discussing potential areas of further research and exploration.

# Chapter 2

## Method

## Introduction

This chapter describes the methodologies employed in this work to map invasive species.

Previous work on mapping invasive plants has shown substantial advancements in detection through the use of innovative methodologies for analyzing remote sensing imagery. Baron and Hill (2020) and Kattenborn et al. (2019) both employed RPAS-acquired imagery for assessing woody invasive species in grasslands and forests, respectively. They emphasized the importance of texture analysis in achieving accurate predictions. On the other hand, Michez et al. (2016), Dorigo et al. (2012), and Du et al. (2021) utilized various remote sensing methods for detecting invasive plants, showcasing the effectiveness of integrating spectral, spatial, temporal characteristics, and gray level co-occurrence matrix (GLCM) texture measures. Du et al. (2021) further highlighted the superiority of object-based analysis over pixel-based methods in classifying wetland plant communities.

Recently, for image classification improvement, multi-scale analyses utilizing a Gaussian pyramid (GP) model have been explored to create features at progressively larger spatial scales. In these analyses, a GP model is applied to create features at increasingly larger scales. This approach marks a novel development in the context of invasive species analysis.

Several studies have explored the use of GP and GLCM separately for feature extraction and classification. These methods have proven to enhance classification accuracy and are particularly useful in multi-resolution spatial analyses. In the studies by Behrens et al. (2018) and Yin and Cui (2021), GP played a pivotal role. Behrens et al. (2018) utilized lower spatial resolution levels of GP to effectively extract terrain attributes from a digital elevation model (DEM), enabling analysis at various scales. Yin and Cui (2021) developed a multi-scale feature extraction and classification approach for hyperspectral images, integrating GP with weighted voting. By breaking down hyperspectral images into multiple scales with GP and applying weighted coefficients based on spectral angle distance, they significantly improved classification accuracy, underscoring the efficacy of multi-scale analysis.

Furthermore, the combination of GLCM-based texture measures with GP-based multi-scale feature creation has shown to significantly improve image classification in several fields, including biomedical imaging, as evidenced by Liu et al. (2022) and Ataky et al.(2023), and other image datasets as noted by Roberti de Siqueira et al. (2013). However, the incorporation of GLCMs and GP for feature creation remains unexplored in remote sensing image analysis. In summary, these studies demonstrate the robust versatility of the GP and GLCM in extracting features and performing classification across various domains and image processing applications.

The subsequent sections of this chapter delve into the application of machine learning (ML), specifically focusing on the two methods that will be used in this work: random forests (RFs) and support vector machines (SVMs). These ML methods will use to develop models that will predict the amount of spotted knapweed cover based on multiscale GLCM features derived using the GP. The following section will provide an overview of machine learning, followed by an introduction to the RF and SVM models that will be employed. Additionally, we will discuss feature creation using GLCMs and GPs, feature selection, hyperparameter tuning, and other pertinent aspects of RF and SVM model building.

## Machine Learning: an Overview

In the realm of remote sensing imagery, ML has surged to the forefront as an indispensable tool, propelled by the exponential growth of big data technologies and high-performance computing capabilities. These advancements have enabled ML to become a critical asset in deciphering complex, data-rich environmental operations.

As Liakos et al. (2018) highlight, a fundamental attribute of ML is its capacity to enable machines to autonomously learn to replicate patterns present in input data, thus circumventing the limitations imposed by traditional programming. This autonomous learning is accomplished by constructing computational models that encapsulate the intricacies of real-world phenomena through the formulation of input-output relationships derived from extensive datasets. Such ML models are particularly skilled at constructing equations to depict these relationships, adeptly handling their potential

nonlinearity and discontinuity. This allows them to uncover intricate patterns that are often too complex for closed-form mathematical equations to represent accurately and may remain undetected by traditional non-ML empirical modeling methods, such as regression analysis.

ML methods can be categorized into broad groups, distinguished by the nature of the learning involved—be it supervised or unsupervised—as well as by the models and specific methodologies employed, such as classification, regression, clustering, and dimensionality reduction (Alpaydin, 2020). Supervised learning requires a set of examples that demonstrate the input-output pattern to be modeled, whereas unsupervised learning infers the output based on the input data alone. Each example is characterized by a set of features serving as predictors for the desired output. If a supervised learning method is used, the set of outputs (e.g., a class label) corresponding to each example is also provided. Utilizing statistical optimization, the parameters of the ML model are refined to enhance performance in solving the particular problem at hand. This iterative optimization and learning phase, known as "training," utilizes a designated set of examples and outputs (for supervised learning only) called "training data" to guide the model development. Once trained, an ML model can not only be used to predict outcomes for new, unseen examples, but also to facilitate a deeper understanding of the underlying data relationships through the examination of its parameters (Alpaydin, 2020; Baştanlar and Özuysal, 2014). This work will employ two supervised classification methods, namely random forest (RF) and support vector machine (SVM).

In addition to the training data, another critical input for ML models is their hyperparameters, which dictate the model's structure and the nuances of its training protocol. These hyperparameters, which are set prior to training, are crucial as they directly affect the model's learning proficiency, effectiveness, and operational efficiency. The efficacy of an ML model depends on various determinants, including the abundance and integrity of the training data, the intricacy of the connections between input and output variables, and practical constraints such as the time and memory resources available for training (Baştanlar and Özuysal, 2014). In this context, the selection and



optimization of hyperparameters are paramount, as they play a pivotal role in harmonizing the model's architecture with the complexity of the task to ensure optimal performance.

The performance of a machine learning model in a specific task is quantified by evaluating the model using a set of examples and associated outputs known as the validation data. These data are held out from the training process and, thus, can demonstrate if the learned model will generalize to new cases. Several performance metrics have been proposed, and this work will use four popular metrics, namely, accuracy, precision, recall, and the F1 score (Murphy, 2012).

- **Accuracy:** This is the ratio of correct predictions to the total number of predictions. It measures the overall correctness of the model.
- **Precision** is the ratio of correctly predicted positive observations to the total predicted positives.

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

where:  $TP$  is the number of true positives.  $FP$  is the number of false positives.

- **Recall** (or Sensitivity or True Positive Rate) is the ratio of correctly predicted positive observations to all actual positives.

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

where:  $TP$  is the number of true positives.  $FN$  is the number of false negatives.

- **F1 Score** is the harmonic mean of precision and recall. It tries to find the balance between precision and recall.

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (9)$$

- **Macro Average** averages the precision, recall, and F1-score for each class independently, treating every class with equal importance. It gives a sense of the model's overall performance across all classes without taking their distribution into account.
- **Weighted Average** takes the class distribution into account by assigning a weight to each class's performance metrics (such as precision and recall) based on how frequently each class appears in the dataset. This metric can offer a more nuanced

view of model performance when the test set mirrors the real-world distribution of classes. Nevertheless, in cases of significant class imbalance, it may over-represent the model's efficacy on the more dominant classes, potentially overestimating the overall performance if the model is better at predicting the majority classes than the minority ones.

- **Support** refers to the number of instances of each class in the dataset, which can directly impact the weighted average and is an important factor to consider when assessing model performance.

These metrics are used together because they reveal different aspects of the model performance, especially in scenarios with imbalanced class distributions, as mere accuracy can be misleading in these situations (Murphy, 2012).

### Data Partitioning: Training and Validation Data

Predictive modeling using ML often involves tuning parameters and possibly feature selection to allow accurate replication of the input/output pattern. To determine optimal parameter settings and relevant features, it is vital to use existing data. In machine learning, the comprehensive data set is typically divided into training and validation subsets. The training set aids in model construction, tuning, and feature selection, while the validation set assesses predictive performance (Kuhn and Johnson, 2013).

The model's performance is influenced by the distribution of output variables in both training and validation sets. An unbalanced training set, where output variables aren't distributed uniformly, can lead to a model bias. This bias makes the model proficient at predicting dominant outputs but weak at predicting less frequent outputs. Similarly, an unbalanced validation set can skew performance measures, emphasizing the dominant outputs' prediction accuracy. For classification models with discrete output variables, stratified random sampling, as opposed to simple random sampling, is advised for partitioning data to training and validation sets (Kuhn and Johnson, 2013). When aiming for balanced training and validation sets, the least represented class often determines the size of the training data set. To ensure diversity in more frequent classes, resampling or bootstrapping is common, generating multiple training and validation set versions. This

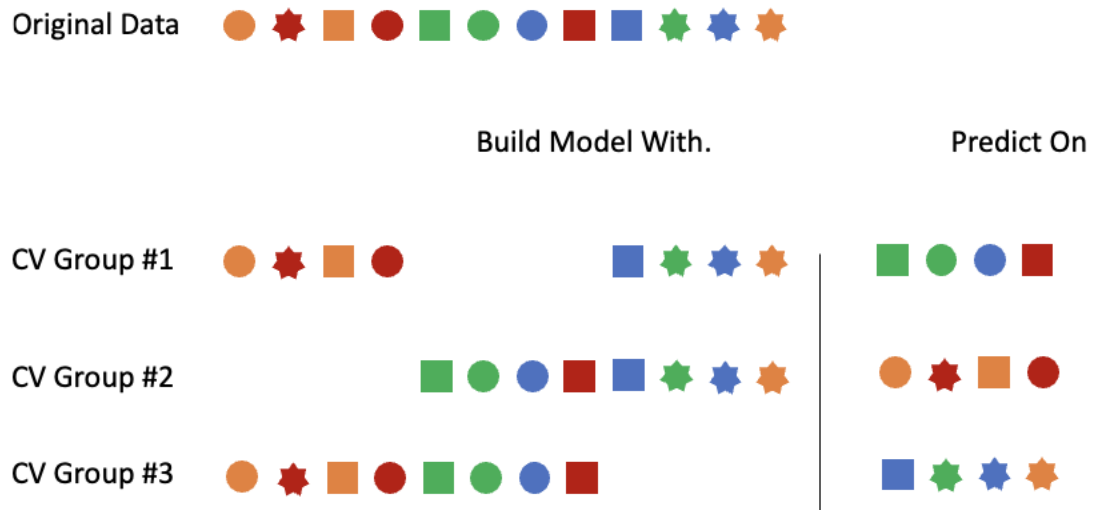
technique has been proven to enhance model performance (Kuhn and Johnson, 2013) and will be used in this work.

## Cross Validation

Cross-validation is a strategy to systematically partition a training set. This division allows a segment of the data to be utilized for model training while the remainder assesses the model's performance, guiding its training by adjusting hyperparameters or halting it if ineffective (Fushiki, 2011). This iterative process ensures all training set examples contribute to both training and model assessment. Beyond mitigating overfitting, cross-validation-derived performance metrics tend to be more reliable than those computed without this procedure (Wong and Yeh, 2020).

### *K-Fold Cross Validation*

K-fold cross-validation divides the training set into 'k' equally sized subsets, or folds. Membership in these subsets is determined through random selection. The model is initially trained using all folds except the first, subsequently predicting the outcomes for the excluded fold to evaluate performance. This cycle repeats, with each fold being excluded once. The outcome is 'k' distinct performance estimates, typically presented as mean and standard error values. This aggregated data provides insights into the influence of model hyperparameters (Kuhn and Johnson, 2013). The cross-validation process with  $k=3$  is depicted in Figure 2.



*Figure 2: A diagram demonstrating threefold cross-validation. Symbols represent training set samples, divided into three groups. Sequentially, each group is excluded during model training. Performance estimates, such as error rate, are derived for each withheld sample set. Averaging these performance metrics gives the cross-validation estimate of model performance.*

The decision of the number of folds ( $k$ ) to use in cross-validation is crucial. While 5 or 10 are standard choices, there's no universal standard. A larger number of folds means the training model uses a sample size nearing the complete training set. For example, when  $k$  equals the total training set size ( $N$ ) - termed "leave-one-out" cross-validation - the model trains on  $N-1$  samples. One sample is reserved in each training cycle for performance estimation. This approach reduces bias, where "bias" is the difference between estimated and true performance values. Thus, a larger  $k$  value, like 10, tends to display less bias than a smaller 'k' value, such as 5 (Fushiki, 2011).

In this study, we applied a dual-validation approach to the machine learning model. Initially, a 10-fold cross-validation on the training dataset was used to optimize the model parameters and ensure its robustness. Each fold cyclically trained and tested the model, facilitating parameter refinement. Subsequently, once the model was developed, an independent validation set, separate from the training data set, was used to evaluate the model's capability to generalize and make accurate predictions in unfamiliar scenarios.

## Recursive Feature Elimination (RFE)

Feature Selection involves sequentially assessing each feature in a dataset to identify their effectiveness on the outcome. The goal is to achieve the same or higher accuracy while reducing the dimensionality of data with many features. The Recursive Feature Elimination (RFE) technique evaluates the classifier's performance by systematically removing features. Features that can be removed without resulting in a significant decrease in model performance are considered to be expendable, while features that cannot be removed without resulting in a significant performance decrease are considered to represent the optimal feature set. RFE begins by considering the full set of features, and then calculates the model's performance. It also assesses the significance or ranking of each feature within the classifier. Through iterative cycles, subsets are generated by progressively eliminating features. In each iteration, the classifier is retrained, its performance reassessed, and the importance or rankings of the remaining features are recalculated (Svetnik et al., 2004; Ustebay et al., 2018).

Various techniques are often utilized for evaluating the model performance during RFE, including Recursive Feature Elimination with Cross-Validation (RFECV) (Misra and Yadav, 2020), Principal Component Analysis (PCA) (Wibawa and Novianti, 2017), and measures like Information Gain Ratio and Information Gain (IG) (Adi et al., 2019). In RFECV the training data are divided into folds, and RFE is performed on each fold. The cross-validation scores of the resulting classifiers are averaged, and the number of features selected for the fold that produces the best score is determined to be the optimal number ( $N_{opt}$ ) of features. Finally, RFE is performed using the entire training dataset to identify the optimal set of  $N_{opt}$  features. Misra and Yadav (2020) have shown that the RFECV method can enhance the performance of classification algorithms. Additionally, Chang et al. (2019) has demonstrated that employing the RFECV algorithm with 10-fold cross-validation significantly improves the precision of algorithms such as RF and Extreme Gradient Boosting (XGBoost), even with a reduced attribute set.

For this study, a two-step recursive feature elimination process was developed using the RFE and RFECV methods in the open-source Python library, SciKit Learn. The first step

RFECV was employed to construct feature sets of all sizes from 1 to  $N$  features, where  $N$  is the maximum number of features available. For each feature set size, the cross-validation score was calculated. This process was repeated to account for stochasticity in the fold-selection and the training process, and the average cross-validation score was calculated. A plot of the average cross-validation score versus the size of the feature set was plotted, and a visual analysis was conducted to identify the optimal feature set size ( $N_{opt}$ ). The feature set size that corresponded to the highest average cross-validation score is selected unless, there are ties. In the case of ties for the highest average cross-validation score, the principle of parsimony was applied, and the smallest feature set size of all of ties was selected. In the second step, RFECV was repeated this time to select which  $N_{opt}$  features to include in the optimal feature set. This selection process was repeated to address stochasticity in the fold selection and model training. Finally, the optimal feature set was defined to be the  $N_{opt}$  features that were selected by the majority of these repeated RFECV runs.

## Random forest

This section delves into the RF algorithm, highlighting the core mechanisms behind the construction of its ensemble of trees. The nuances governing the tree-building process are explored, alongside the essential aspect of hyperparameter tuning in optimizing the RF model's performance. The RF, a supervised machine learning model, is implemented in this study using the open-source Python library, SciKit Learn. An RF comprises an ensemble of decision trees (Breiman, 2001; Mingers, 1989). Decision trees use a hierarchically partition a data from the root node down to the leaf nodes as depicted in Figure 3.

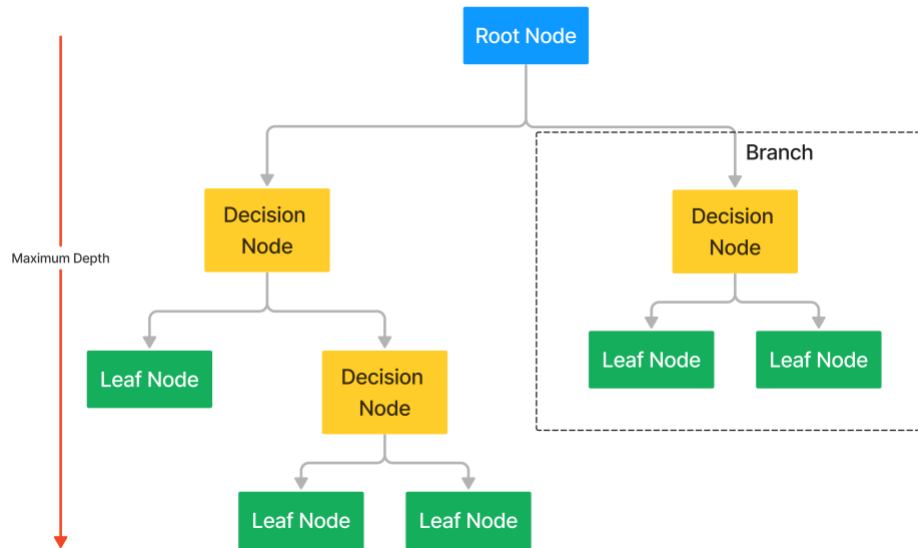


Figure 3: Decision tree structure showcasing root nodes, decision nodes, leaf nodes, and branches.

Decision trees are differentiated into classification trees, which are designed for qualitative response variables, and regression trees, tailored for quantitative response variables (Strobl et al., 2009). A known characteristic of decision trees is their sensitivity to the training data; even minor changes in the data can significantly alter the tree's structure, potentially leading to poor generalization in the resulting classifier (Sheykhmousa et al., 2020). To enhance generalization, the predictions from multiple trees can be aggregated in an ensemble method. The RF algorithm adopts this strategy by fitting numerous individual trees, often hundreds or thousands, and integrating their predictions to produce a final outcome (Elith, 2019; Pal, 2005; Strobl et al., 2009). The model user must define the number of trees, a hyperparameter, which contributes to mitigating the over-specificity of single trees and leads to more reliable predictions (Hastie et al., 2009).

In RF, each tree is constructed from a bootstrap sample of the training data, drawn with replacement, equal in size to the training dataset. On average, these samples contain about 63.2% unique records, the so-called in-bag samples, while the remaining unchosen records, the out-of-bag samples, help estimate the model's error rate (Efron and Tibshirani, 1994). RF differentiates itself from other ensemble methods such as bagging

by introducing randomness in the selection of predictors at each split, evaluating only a subset of predictors to find the optimal one. This strategy generates decorrelated trees and reduces overfitting risks (Breiman, 2001; Elith, 2019; Strobl et al., 2009).

RF typically constructs "deep" trees, unpruned with many splits, which may lead to terminal nodes that have few data points. The minimum sample size for terminal nodes and the tree depth are adjustable through hyperparameters, which vary in their specific control features according to the software implementation (Elith, 2019; Strobl et al., 2009; Valavi et al., 2021).

Once trained, the RF model applies its collective knowledge to new, unlabeled input data. Each decision tree in the ensemble contributes a vote towards a class membership for each data point. The class that garners the majority of votes is assigned as the prediction for the given input (Breiman, 2001; Klusowski, 2018).

### Optimizing RF performance: Hyperparameters Tuning

The RF is a flexible and potent algorithm but demands careful calibration of various hyperparameters to deliver optimal performance. The appropriate choice of hyperparameters can significantly affect the accuracy, generalizability, and efficiency of the RF model. These include the number of observations selected at random for each tree and whether they are drawn with or without replacement, the count of variables chosen randomly for each split, the criterion for splitting, the minimum required samples within a node, and the total number of trees in the ensemble (Probst et al., 2019).

A significant part of executing the RF model is the configuration of its two primary hyperparameters: the number of trees (*Ntree*), and the number of randomly selected features to be considered at each split (*Mtry*). The number of trees in a RF impacts the model's ability to represent patterns characterized by a large number of training data or a large number of features in each training example. As the training set size or number of features increases, so too should the number of trees in the RF. Each tree in the RF has the potential to use any of the predictive features for classifying an example. However, each split in the tree will be selected using a unique subset of these predictors. The size of this subset is defined *Mtry* parameter. Berhane et al. (2018) pointed out, while the RF



model's behavior is generally resilient to changes in *Ntree*, it can be more sensitive to variations in *Mtry*. Reducing the *Mtry* parameter can lead to quicker training, but it also diminishes both the correlation between any two trees and the individual strength of each tree in the forest. Consequently, the value of this hyperparameter has a multifaceted impact on classification accuracy (Klusowski, 2018).

Given that the RF classifier is computationally efficient and tends not to overfit, it can accommodate a very large number of trees (*Ntree*) (Guan et al., 2013). However, numerous studies have identified 500 as an optimal number for *Ntree*, as further increases did not yield improvements in accuracy (Belgiu and Drăguț, 2016). In contrast, the ideal value for *Mtry* depends on the specific dataset. For classification tasks, it is advised to set the *Mtry* parameter to the square root of the number of input features (Breiman, 2001).

This study uses the RF implementation in the Python's scikit-learn library for Python. In this RF implementation, *the hyperparameters Mtry and Ntree* are named *max\_features* and *n\_estimators*, respectively. The complete set of hyperparameters in the scikit-learn RF implementation are:

- *Max\_depth*: Each tree's depth in the forest determines its complexity. By setting an explicit max depth, we can prevent the tree from growing endlessly. An overly deep tree might lead to overfitting; conversely, a shallow tree might underfit the data.
- *Min\_samples\_split*: This refers to the minimum count of data points placed in a node before the node is split. For instance, if the value is set to 10, a split will be attempted only if it contains at least 10 data points.
- *Min\_samples\_leaf*: After a successful split, nodes might contain a small count of data. If this count is lesser than 'min\_samples\_leaf,' then the split will be deemed unsuccessful. This is a regularization hyperparameter that helps in avoiding overly specific leaves in the tree.
- *Bootstrap*: A core concept in the RF model, bootstrapping involves training each tree on a distinct subset of data. This subset, known as the "bag," is sampled with replacement from the entire dataset. The data points not in the bag are called

"Out of Bag (OOB) samples." Aggregating the outputs from multiple such diversified trees helps in reducing the model's variance and increases its robustness (Kelkar and Bakal, 2020).

These hyperparameters are critical to RF modeling and will be tuned to find the optimal set of hyperparameters for a given dataset (Probst et al., 2019).

## Support Vector Machine (SVM)

First proposed by Vapnik and his team in the late 1970s, the SVM, a supervised non-parametric statistical learning technique, has become one of the most prevalent kernel-based learning methods in diverse machine learning tasks, notably in image classification (Vapnik, 2006). Fundamentally SVM is a linear binary classifier that delineates a singular boundary between two categories. This linear SVM presupposes that the multi-faceted data can be linearly divided in the input domain (as illustrated in Fig. 2-3). Specifically, SVMs determine an optimal hyperplane (i.e., a surface in the dimensional space defined by the input variables) to divide the dataset into specific pre-established classes, based on training data. To ensure the largest separation or margin, SVMs utilize a subset of the training data that is nearest in the feature space to the best optimal boundary, known as the support vectors (Foody and Mathur, 2004).

The optimal boundary, often termed the "maximal margin" or "optimal hyperplane," is a key aspect of SVM. It represents a decision-making border designed to minimize errors when categorizing data during the training phase (Mountrakis et al., 2011). Referring to Figure 4, multiple hyperplanes are chosen such that no data samples lie between them. The best hyperplane is identified by maximizing the separation distance between these hyperplanes. This systematic approach is termed the learning process.

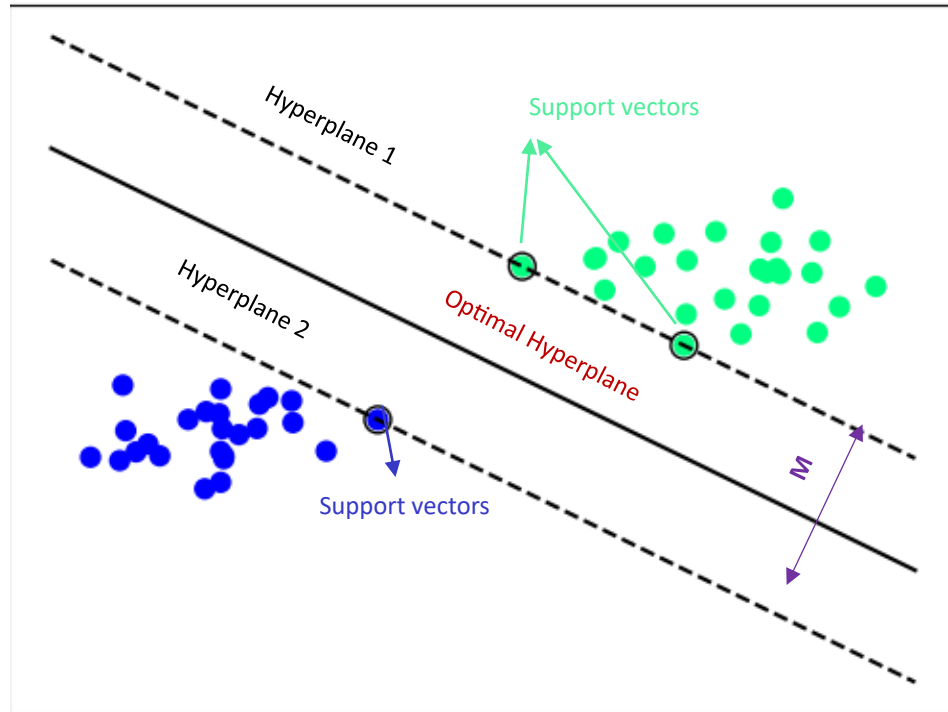


Figure 4: Illustration of an SVM classifier with a linear kernel applied to linearly separable data, highlighting the optimal separating hyperplane (solid line) and the margins (dashed lines) defined by the support vectors (circled points).

In real-world scenarios, data points from different classes may not always be cleanly separated, leading to overlaps, as depicted in Figure 5. Recognizing these challenges, Cortes and Vapnik (1995) introduced significant enhancements to SVM, notably the soft margin and the kernel trick. The soft margin method introduces slack variables to the SVM optimization process, allowing some linear data separation. Concurrently, the kernel trick aims to transform the original data into a higher-dimensional space, making previously overlapping samples more distinct, as shown in Figure 6. The efficacy of SVM largely depends on the right choice of kernel function. Commonly used kernel functions include the Sigmoid, Radial basis function, Polynomial, and Linear models (Cherkassky and Ma, 2004). Specifically, the polynomial and radial basis function (RBF) kernels are frequently used in analyzing remotely sensed images (Mountrakis et al., 2011).

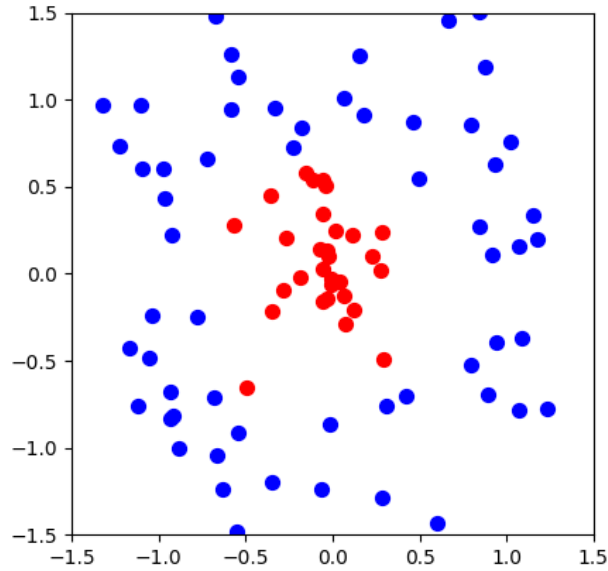


Figure 5: Visualization of a non-linear SVM classification, showing how a non-linear boundary effectively separates two distinct classes in the feature space.

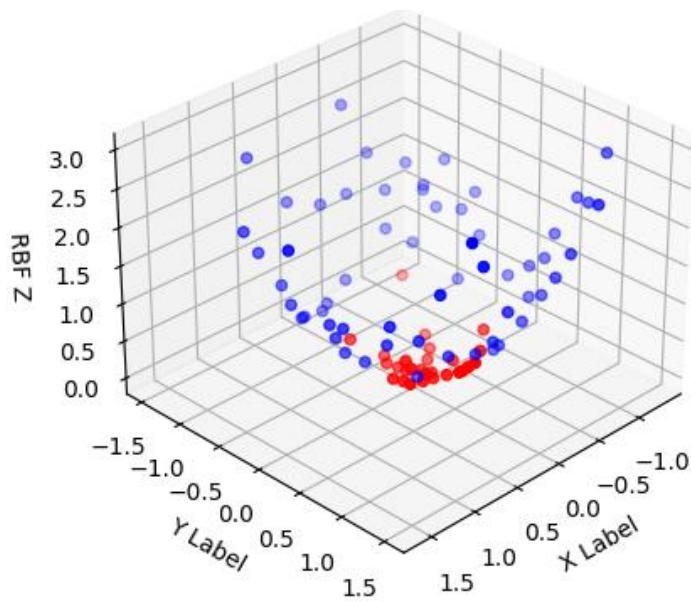


Figure 6: Illustration of the kernel trick applied to data in a 3D space, demonstrating the transformation that facilitates the separation of classes which are not linearly separable in the original dimensions.

## Optimizing SVM Performance: Hyperparameters Tuning

Parameters, including the kernel's parameters  $C$ ,  $\gamma$ , the degree, and the kernel function itself, require meticulous optimization.

- **C (Cost parameter):**

$C$  is a regularization parameter in SVM. It determines the trade-off between achieving a low error on the training data and maintaining a wide margin between classes. A small value of  $C$  creates a wider margin, which may result in more training errors. A large value of  $C$  aims for a smaller margin and fewer training errors. However, setting  $C$  too high might make the model overfit to the training data, reducing its ability to generalize to unseen data.

- **$\gamma$  (Gamma):**

$\gamma$  is a parameter specific to the Radial Basis Function (RBF) kernel. It determines how far the influence of a single training sample reaches, implying how closely the model will fit to the training data. A low  $\gamma$  value makes the model more flexible by considering a broader range of influence for each training sample, producing a more generalized solution. A high  $\gamma$  value considers only close points, producing a more fitted solution, but with a risk of overfitting.

- **Degree:**

This parameter is specific to the polynomial kernel in SVM. It sets the degree of the polynomial function used, altering the complexity of the decision boundary. A higher degree polynomial can capture more complex relationships in the data but increases the risk of overfitting if the complexity is not warranted by the data structure.

These hyperparameters are pivotal in defining the decision boundaries of SVM classifiers and are essential in the model's capacity for generalization. The careful tuning and in-depth understanding of these parameters are critical, as they have a profound effect on the SVM's performance with both training and unseen datasets. This research incorporates the aforementioned hyperparameters for model tuning, with detailed values and results to be presented in Chapter 3.

## Optimizing Machine Learning Performance: Grid Search

To ensure the highest performance of our machine learning models, it is crucial to find the right combination of hyperparameters. In this work uses a grid search method, as detailed by Ataei and Osanloo (2004) and Probst et al. (2019). Hyperparameter tuning via grid search is an exhaustive method that systematically evaluates the training performance of the machine learning model across every possible combination of hyperparameter values provided. The combination of hyperparameters that yields the best training performance is selected as the optimal set for the model (Ataei and Osanloo, 2004).

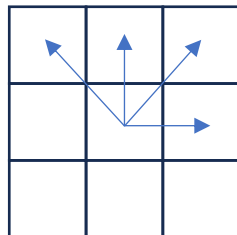
Although this meticulous approach is time consuming, its simplicity to implement, and exhaustive nature makes it a preferred choice for many researchers and practitioners. This is because it provides confidence that the selected hyperparameters are, indeed, the best among the provided set, leading to reliable and robust machine learning model performance (Probst et al., 2019)

## Gray Level Co-occurrence Matrix (GLCM) Based Texture Analysis

There are many methods proposed for extracting textural features in texture analysis. One such method, which is utilized in this research, is the gray level co-occurrence matrix (GLCM) (Roberti de Siqueira et al., 2013). In image processing, especially when delving into the intricate domain of feature extraction, understanding the GLCM becomes crucial. This technique has formed the backbone of many significant advancements in this field (Hall-Beyer, 2017; Mohanaiah et al., 2013; Öztürk and Akdemir, 2018; Xian, 2010; Zulpe and Pawar, 2012).

At the most basic level, a GLCM encapsulates spatial patterns within a black-and-white image. These spatial patterns can be summarized using statistics to create textural features that can aid in image analysis, such as the classification of multispectral images. Haralick et al. (1973) introduced GLCM-based texture features, and since then, they've found use in many remote sensing image analyses, especially for detecting invasive plant species (Baron et al., 2018; Baron and Hill, 2020; Dorigo et al., 2012; Li et al., 2019; Pearlstine et al., 2005).

A GLCM is a square matrix that provides insights into the spatial distribution of gray levels within an image. The term gray-level refers to the pixel value in a single-spectral-band image (i.e., a black-and-white image). The number of rows and columns in a GLCM is defined by the number of gray levels in the image. Thus, an image where the gray tone is encoded as an unsigned integer will have 256 rows and 256 columns. Each element of a GLCM represents the relative frequency of occurrences of two pixels, one with gray-tone  $i$  and the other with gray-tone  $j$ , separated by a distance  $d$  and oriented according to the angular relationship ( $a$ ) to each other. Thus, the co-occurrence pattern, captured by the GLCM, is dependent on both the separation distance ( $d$ ) and angular relationship ( $a$ ) between the pixels. The angular relationship between pixels can be defined by four different directions across an image: horizontal, vertical, left diagonal, and right diagonal, as illustrated in Figure 7. Because these angular relationships are reciprocal, the resultant GLCM is symmetric. This means patterns observed in one direction (e.g., up) are similar to its counterpart (e.g. down) forming a harmonized relationship, whether horizontal, vertical, or diagonal.



*Figure 7: Illustration of the four primary directions (0°, 45°, 90°, and 135°) used in the computation of Haralick texture features for GLCM analysis, depicting the relative positioning of pixel pairs.*

Within this framework, various textural features emerge, and this study predominantly focuses on five: Angular Second Moment (ASM), Correlation (COR), Entropy (ENT), Sum Entropy (SENT), and Difference Entropy (DENT). These features were selected because they are invariant to gray-tone transformations (Gonzalez et al., 2008; Haralick et al., 1973), and thus, are expected to be less sensitive to changes in illumination (e.g. shadows) and calibration errors.

- ASM quantifies the uniformity of the distribution of gray levels separated by a distance  $d$  in the direction  $a$  in the image.
- COR is a measure that indicates the presence of linear dependencies between gray levels separated by a distance  $d$  in the direction  $a$  in an image. It provides insights into the relationship between the rows or columns of the GLCM and their degree of association with each other (Connors and Harlow, 1980; Kekre et al., 2010; Tahir et al., 2003).
- ENT is calculated by assessing the probability of occurrence of a pixel with a certain intensity next to a pixel with another intensity. A high entropy value from the GLCM indicates a high degree of complexity and variability in the image texture, suggesting a less uniform and more detailed pattern. Conversely, a low entropy value implies more homogeneity and less detail, reflecting a more predictable texture pattern (Baraldi and Pannigiani, 1995; Haralick et al., 1973; Haralick and Shanmugam, 1973).
- SENT assesses the complexity in an image by examining the sum of intensities of pixel pairs. Higher values indicate more complexity or texture information in the image (Haralick et al., 1973).
- DENT is a measure of the variability in the differences between the gray levels of pixel pairs. Instead of looking at how often pairs of gray levels occur together, as with traditional GLCM entries, Difference Entropy examines the frequency distribution of the absolute differences between the gray levels of each pixel pair. It calculates the entropy of this difference distribution, capturing the texture's contrast. A high Difference Entropy value indicates a greater complexity or variability in texture contrast, while a low value suggests less contrast and more uniformity in the texture of the image (Haralick et al., 1973).

This analytical method allows for a deeper understanding of image content, facilitating advancements in areas like medical imaging, remote sensing, and even automated quality inspection in manufacturing (Roberti de Siqueira et al., 2013).



## Mathematical Representation of GLCM-based Features

These features can be calculated using the equations below:

### Distribution of Gray Levels along the Horizontal Axis:

$$p_x(i) = \sum_{j=1}^{N_g} p(i, j) \quad (10)$$

where  $i$  is a horizontal position in the image.  $j$  is a particular gray level intensity value.  $N_g$  is the number of possible gray level intensities.  $p(i, j)$  represents the probability of finding a pixel with gray level  $j$  at the horizontal position  $i$ .  $p_x(i)$  represents an average gray level intensity at horizontal position  $i$ , considering all vertical positions in the image.

### Mean of $p_x$ :

$$\mu_x = \frac{1}{N_g} \sum_{i=1}^{N_g} p_x(i) \quad (11)$$

### Standard deviation of $p_x$

$$\sigma_x = \sqrt{\frac{1}{N_g-1} \sum_{i=1}^{N_g} (p_x(i) - \mu_x)^2} \quad (12)$$

### Distribution of Gray Levels along the Vertical Axis

$$p_y(j) = \sum_{i=1}^{N_g} p(i, j) \quad (13)$$

where  $p_y(j)$  represents an average gray level intensity at vertical position  $j$ , considering all horizontal positions in the image.

### Mean of $p_y$ :

$$\mu_y = \frac{1}{N_g} \sum_{j=1}^{N_g} p_y(j) \quad (14)$$

Because of the assumption of symmetry in the angular relationships considered in this work,  $\mu_x = \mu_y$ .

**Standard deviation of  $p_x$ :**

$$\sigma_y = \sqrt{\frac{1}{N_g-1} \sum_{i=1}^{N_g} (p_y(i) - \mu_y)^2} \quad (15)$$

Because of the assumption of symmetry in the angular relationships considered in this work,  $\sigma_x = \sigma_y$ .

**Distribution of Gray Level Sum:**

$$\begin{aligned} p_{x+y}(k) &= \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i, j) \\ k = i + j \quad k &= 2, 3, \dots, 2N_g \end{aligned} \quad (16)$$

$p_{x+y}(k)$  represents the probability distribution of the sum of gray level intensities of pixels at positions  $i$  and  $j$ , and signifies the likelihood of encountering a specific sum of gray levels in the image.

**Distribution of Gray Level Difference:**

$$\begin{aligned} p_{x-y}(k) &= \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i, j); \\ k = |i - j| \quad k &= 0, 1, \dots, N_g - 1 \end{aligned} \quad (17)$$

$p_{x-y}(k)$  represents the probability distribution of the difference in gray level intensities of pixels at position  $i$  and  $j$ .

**Angular Second Moment (ASM):**

$$ASM = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \{p(i, j)\}^2 \quad (18)$$

where  $p(i, j)$  represents the probability (or normalized frequency) that a pixel with gray level  $i$  is adjacent to a pixel with gray level  $j$  in a particular direction (e.g., horizontal, vertical, diagonal). Squaring the probabilities emphasizes higher probabilities, which, in the context of the ASM, it gives a measure of uniformity.

**Correlation (COR):**

$$COR = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i * j) p(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (19)$$

**Entropy (ENT):**

$$ENT = - \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i, j) \log_2 [p(i, j) + \varepsilon] \quad (20)$$

where  $\varepsilon$  is a small constant value to ensure that the logarithm is defined (preventing log of zero, which is undefined), which ensures stability in the computation, especially for cases where  $p(i, j)$  might be zero.

**Sum Entropy (SENT):**

$$SENT = - \sum_{i=2}^{2N_g} p_{x+y}(i) \log_2 [p_{x+y}(i) + \varepsilon] \quad (21)$$

where  $\varepsilon$  is a small constant value to ensure that the logarithm is defined (preventing log of zero, which is undefined), which ensures stability in the computation, especially for cases where  $p(i, j)$  might be zero.

**Difference Entropy (DENT):**

$$DENT = - \sum_{i=0}^{N_g-1} p_{x-y}(i) \log_2 [p_{x-y}(i) + \varepsilon] \quad (22)$$

where  $\varepsilon$  is a small constant value to ensure that the logarithm is defined (preventing log of zero, which is undefined), which ensures stability in the computation, especially for cases where  $p(i, j)$  might be zero.

In the scope of this research, the focus remains on a separation distance  $d = 1$  for deriving textural features. To represent these directional features in an all-encompassing, rotation-invariant manner, each of the four directional (vertical, horizontal, diagonal-up, diagonal-down) variations is summarized using mean and range statistics (Sebastian et al., 2012).

## Introduction of Scale-Space Theory and the Gaussian Pyramid

Scale-space theory stands as a fundamental pillar in the realm of image processing, enabling a multi-scale representation of images, particularly when interpreting multi-

spectral imagery using a Gaussian pyramid (GP). The essence of this research revolves around understanding how multi-scale attributes play a pivotal role in image classification.

## Importance of Multi-scale Analysis

Objects and landscapes present in our environment inherently possess multi-scale characteristics. Depending on the observation scale, these objects might exhibit varying appearances. Similarly, biological vision systems display different levels of visual processing, each corresponding to a specific scale of information. As automated algorithms for image interpretation in novel scenes evolve, the challenge of determining relevant scales, without prior knowledge, becomes paramount. To address this, scale-space theory advocates for simultaneous representation across all scales, offering a structured methodology to represent an image through a series of smoothed or blurred versions spanning different scales (Florack et al., 1992; Koenderink, 1984; Lindeberg, 2009, 1995, 1990; Romeny, 2008; Witkin, 1983).

## Axiomatic Derivations and the Gaussian Approach

A core tenet of scale-space theory, based on axiomatic derivations, posits that representations at coarser scales should ideally be simplified versions of their finer-scale counterparts (Lindeberg, 2013). Such a guideline naturally suggests a specific class of image operators: convolution using Gaussian kernels and their derivatives. These operators not only capture varying scale information but also retain pertinent image structures (Lindeberg, 2020; Mikolajczyk, 2002). The Gaussian-based approach, thus, emerges as an efficient and robust technique for a wide array of visual processing endeavors. Its application ranges from feature detection, classification, image-based pattern recognition, and image segmentation to enhancement via deblurring (Kuijper et al., 2003).

With its foundations deeply embedded in both physics and biological vision, scale-space theory provides a cohesive methodology for computer vision. By offering a systematic, multi-scale analysis tool, this theory has gained traction and widespread application in various computer vision tasks. As technology continues to evolve, the principles of scale-space theory will undoubtedly remain integral in shaping future

developments in the field of image processing and interpretation (Chomat et al., 2001; Florack, 1997; Henkel, 1995; Hummel et al., 1987; Kalitzin et al., 1997; Réti, 1995).

## Gaussian Convolution and Scale-Space

Gaussian convolution plays a central role in the creation of scale-space representations by enabling the construction of a set of multi-scale images that capture image structures at various levels of granularity. The process involves applying a Gaussian kernel to the original image at different scales. This results in a series of increasingly smoothed versions of the image, each representing a different level of blurriness (Kuijper et al., 2003). The two-dimensional Gaussian function used to define a kernel is:

$$g(x, y; \sigma, \mu_x, \mu_y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-\mu_x)^2 + (y-\mu_y)^2}{2\sigma^2}\right) \quad (23)$$

where,  $\sigma$  is the standard deviation of the Gaussian function,  $\mu_x$  is the x-coordinate of the center of the Gaussian function.  $\mu_y$  is the y-coordinate of the center of the Gaussian function and  $1 / (2\pi\sigma^2)$  is the normalization factor to ensure that the integral of the Gaussian function over the entire domain is equal to 1,  $(x - \mu_x)^2 + (y - \mu_y)^2$  is the squared distance from the origin, and the negative sign ensures that the function decreases as the distance from the origin increases.

In Figure 8, a 2D Gaussian distribution is depicted with  $\mu_x = 0$ ,  $\mu_y = 0$ , and  $\sigma=1$ .

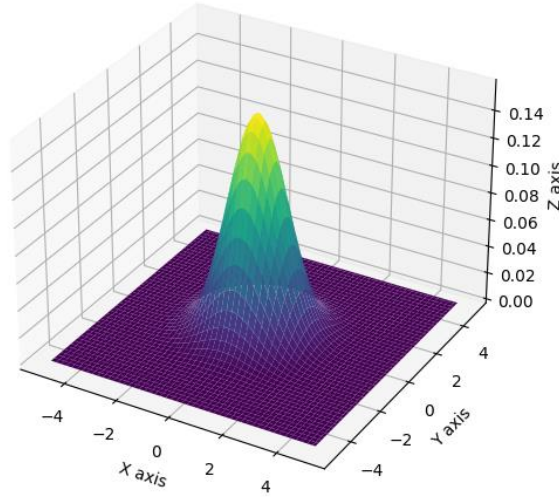


Figure 8: 3D representation of a 2D uniform Gaussian kernel, visualizing the symmetrical distribution and peak concentration at the origin.

The family of Gaussian kernels has several properties that facilitate its use for data smoothing, namely, linearity, separability, causality, and the semi-group property (Florack et al., 1992; Koenderink, 1984; Lindeberg, 1995; Sporring et al., 2013; Witkin, 1983). Of particular relevance is the property of separability, which allows an  $n$ -dimensional Gaussian kernel to be derived as the product of  $n$  one-dimensional kernels (Mikolajczyk, 2002). This property can be represented mathematically as:

$$g(x,y) = g(x)g(y) \quad (24)$$

$$\text{where, } g(x) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{(x-\mu_x)^2}{2\sigma^2}} \text{ and } g(y) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{(y-\mu_y)^2}{2\sigma^2}}$$

The separability property simplifies the computational complexity of convolutions and contributes to efficient image processing algorithms because it allows the smoothing of an image to be accomplished through two separate one-dimensional smoothing steps, each applied to one dimension of the image.

Typically, the creation of different levels in the scale-space representation involves convolving the image with the Gaussian kernel.

$$L(p,\sigma) = g(\sigma) * I(p) \quad (25)$$

where  $*$  is the convolution, with  $I$  the image and  $p = (x,y)$  the point location. The Gaussian kernel,  $g$ , is characterized by circularly symmetric and is parameterized by a

single scale factor, denoted as  $\sigma$ . Using the separability property of Gaussian kernels permits a two-dimensional Gaussian kernel to be decomposed into two orthogonal, one-dimensional filters, which significantly reduces the computational complexity (Mikolajczyk, 2002). Furthermore, the implementation of a one-dimensional Gaussian kernel can be achieved using a recursive filter (Deriche, 1993). This recursive approach offers notable computational efficiency, particularly when dealing with larger Gaussian kernels (e.g., kernels which operate on a wide areas of neighbouring pixels) (Mikolajczyk, 2002).

## Gaussian Pyramids

The GP is a foundational tool for creating multi-scale representations of images which has been widely used in the realm of image processing and computer vision (Haddad and Akansu, 1991; Konlambigue et al., 2018; Li et al., 2018; Olkkonen and Pesola, 1996, 1996; Sporring et al., 2013). These pyramids have been hailed for their efficiency and wide applicability in various tasks like image compression, segmentation, and object detection (Adelson et al., 1984; Mpinda Ataky et al., 2020; Olkkonen and Pesola, 1996).

The primary goal of a Gaussian pyramid is to facilitate the analysis of images at different resolutions. This is achieved by constructing copies of an image at different levels of detail and scale. The pyramid itself is constructed by stacking images with varied resolutions: starting with the original image at the base and progressively scaling it down to the top, which becomes a single-pixel representation indicating the average value of the entire image (see Figure 9). This layered representation not only reduces noise in the image as pyramid levels increase but also enhances its smoothness, making it invaluable for various image processing applications (Li et al., 2018).

Constructing such a pyramid involves a sequence of operations. Initially, the image undergoes convolution with a Gaussian kernel, which, is typically centered around the center of a pixel or a group of pixels. The standard deviation of the kernel dictates the degree of image blurring (Binaghi et al., 2003; Chaudhuri and Marron, 2000). Post-blurring, the image is then subjected to downsampling. Here, every 2x2 block of pixels in the image is averaged, but this doesn't alter the image's spatial extent. Instead, this

method decreases the image's resolution by creating pixels with spatial footprints that are four times larger. The culmination of this process is the halving of the image's size, transitioning an  $M \times N$  image to an  $M/2 \times N/2$  version, thus cutting down its area to one-fourth of its original size, a process often termed as an octave or level (Li et al., 2018). This process is visually depicted in Figure 9, illustrating the progression from the original image at  $l_0$  to the fourth level of the Gaussian Pyramid ( $l_5$ ).

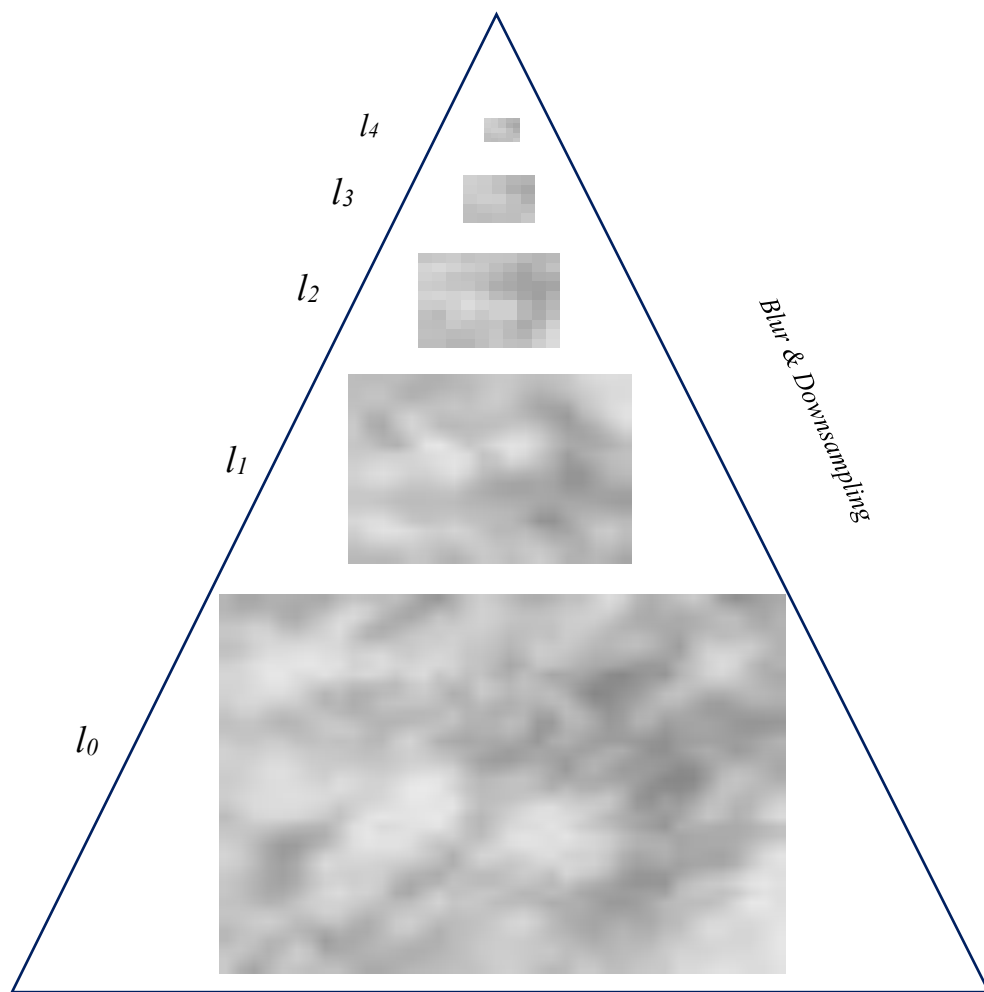


Figure 9: Gaussian Pyramid. The image illustrates five levels of the GP, spanning from the original image at level  $l_0$  to the fifth level,  $l_4$ .



As depicted in Figure 9, images of varying resolutions can be visualized as a stacked structure, called a pyramid (Adelson et al., 1984). At the base of this structure is the original image ( $G_0$ ), which has the highest spatial resolution of all the images in the pyramid. Whereas the lowest resolution image ( $G_N$ ) is found at the apex of the pyramid. Each image ( $G_l$ ) in the pyramid is referred to as inhabiting a level ( $l$ ) in the pyramid, where the zeroth level corresponds to the original image at the base of the pyramid. As the level in the pyramid increases, the image resolution decreases, culminating in the  $N$ th level, which corresponds to the lowest resolution image at the apex. To create the image at the  $l$ -th level in the pyramid, the image at the  $(l - 1)$ -th level is convolved with the Gaussian kernel and the resulting blurred image is then resized. Thus,  $G_l(x, y)$  is an evolution of  $G_{l-1}(x, y)$ , each step seamlessly transforming the image's resolution.

$$G_l(x, y) = \sum_{m=-T}^T \sum_{n=-T}^T w(m, n) * G_{l-1}(2R_x x + m, 2R_y y + n) \quad (26)$$

In this equation:

- $G_l(x, y)$  signifies the pixel value at location  $(x, y)$  in the  $l$ -th level of the Gaussian pyramid.
- $G_{l-1}(2R_x x + m, 2R_y y + n)$  represents the pixel value from the previous level ( $l-1$ ) of the Gaussian pyramid.

The terms  $2R_x x + m$  and  $2R_y y + n$  adjust coordinates for downsampling with  $R_x$  and  $R_y$  being the respective scaling factors for the  $x$  and  $y$  axes.

- The weight  $w(m, n)$  is derived from a predefined Gaussian kernel, emphasizing the importance of each pixel's proximity. It functions as a smoothing filter, reducing rapid pixel value changes across the image. When convolved with an image, it averages pixel values within its domain, generating a blurring effect.
- $T$  determines the limit or the radius of the kernel in each direction. If  $m$  and  $n$  range from  $-T$  to  $T$ , the kernel would be of size  $(2T+1) \times (2T+1)$ .

The symbol  $*$  is the convolution operation, while the downsampling ratios in the  $x$  and  $y$  directions are denoted as  $R_x$  and  $R_y$ , respectively.

Digital images discretize space into pixels, thus, to create a GP, the Gaussian kernel must be discretized. The Gaussian kernel used in this work is represented as (Bradski, 2000):

$$\frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

Each value in the matrix corresponds to a weight, and when this kernel is convolved with an image, it gives a weighted average of the pixel values in the neighborhood defined by the kernel. This kernel gives more weight to the center pixels and increasingly less weight is given as the distance from the kernel center increases, leading to a Gaussian or bell-curve distribution of weights. The factor  $\frac{1}{256}$  normalizes the kernel, ensuring that the sum of all weights is 1.

Figure 2-6 shows an example of a Gaussian pyramid constructed from a single spectral-band (green) image, which was captured by an RPAS at the Laurie Guichon Memorial Grasslands Interpretive Site. Level 0 ( $l_0$ ) in this pyramid corresponds to the original grey-scale image. The subsequent levels are constructed by applying equation 24 and downsampling, which involves reducing the number of pixels by removing even rows and columns. Thus, to create the first layer,  $l_1$ , the Gaussian kernel is applied to the original image ( $l_0$ ). The result is then downsampled so that each pixel represents an area 4x larger on the ground. Here,  $l$  is an element of the set  $\{0,1,2,3, \dots, N\}$ , where  $l$  represents the number of layers in the Gaussian pyramid.

## Conclusion

In this chapter, the methods that are central to the research have been discussed. Moving forward, the approach will involve using RF and SVM classifiers, both of which are trained using a 10-fold cross-validation method. Hyperparameters for these models will be identified using grid-search. To enhance the spectral information present in the RPAS-acquired imagery, GLCM texture features will be created. Additionally, GPs will

be employed to achieve a multi-scale representation of the images, enabling a thorough analysis across different scales. Features will be selected from these GP representations of the RPAS-acquired imagery using the two-step RFECV method described above. The subsequent chapter will detail how these methods are applied to map spotted knapweed in a grassland ecosystem.

# Chapter 3 Invasive Species

## Mapping: A Case Study on Spotted Knapweed Detection in Grassland Ecosystems

## Introduction

The primary objective of this work is to evaluate the impact of image spatial resolution when mapping spotted knapweed in a grassland ecosystem. The data used for this study were collected using a consumer-grade remotely piloted aircraft system (DJI Phantom 4) and multispectral imager (Parrot Sequoia). Processing the raw imagery using Pix4D Pro (Pix4Dmapper, 2018) resulted in four images with a spatial resolution of 2.9 cm and representing the green, red, red-edge, and near-infrared spectral bands. From these images, three vegetation index (NDVI, reNDVI, and gNDVI) images (also with a 2.9 cm spatial resolution) were created. These datasets are described in greater detail in Section 1.8 of this thesis and in Baron (2020).

This chapter will discuss how Gaussian pyramids (GPs) were used to create a multi-scale representation of these seven images, how features were extracted from these images, and how these features were used to create machine-learning classifiers using both random forests (RF) and support vector machines (SVM). The outcome of classifiers built with features calculated from different spatial resolution images will be analyzed to reveal scale dependencies in feature sets. Finally, classifiers built using features from all spatial scales will be compared to evaluate the importance of multiscale feature sets in this classification task. A key aspect of this discussion is the comparison of the data processing framework developed in this work, which includes scale optimization, with the framework employed in previous work (Baron and Hill, 2020), highlighting the intention of this study to evaluate scale-space feature analysis.

## Methods

All analyses were conducted using PyCharm 2023.2.2, an Integrated Development Environment (IDE) specifically for the Python programming language, which facilitated the implementation of image processing through the Scikit-Image 0.20.0 libraries (Van der Walt et al., 2014) and machine learning, specifically Random Forest (RF) and Support Vector Machine (SVM), via the Scikit-Learn 1.2.2 libraries (Pedregosa et al., 2011). GP images were created using the OpenCV library 4.5.4.58 (Bradski, 2000). Data management and analysis were also performed with Pandas 1.4.0, an open-source Python library

renowned for its high-performance and user-friendly data structures and data analysis tools (McKinney et al., 2010). Numerical computations, particularly on arrays and matrices, were handled by NumPy 1.21.4, a Python module known for its rapid computational capabilities (Harris et al., 2020). The Mahotas library, which is dedicated to computer vision and image processing, was utilized to compute the Gray Level Co-occurrence Matrix (GLCM) (Coelho, 2013). Finally, visualization and mapping of the results were done using ArcMap 10.7.2.

## Feature Extraction

### *Gaussian Pyramids*

The GP of an image consists of a series of increasingly blurred images constructed from the original image. The original image constitutes level 0 in the image pyramid, and at each higher level in the pyramid, the image size resolution decreases while the degree of blurring increases. This operation was performed four times to generate four different levels of the Gaussian pyramid, starting at level-0 ( $l_0$ ), which constitutes the original image (40x40 pixels), to level-4 ( $l_4$ ), the fifth level of the GP, which has 3x3 pixels. Figure 10 shows the Gaussian pyramid constructed for a green band image used in this study.

The GP method was applied to the four spectral bands (Green, Red, NIR, Red-edge) and three multiband vegetation indexes (NDVI, gNDVI, reNDVI) images using the OpenCV library (Bradski, 2000). This created a set of 7 image GPs, with 5 levels each. Level-0 of each of these GPs is populated by the spectral image produced from the RPAS-acquired data by Pix4D (or a multiband vegetation index based on these data), and has a ground sampling distance (e.g., the on-the-ground footprint of a pixel) of 2.9 cm. For each successive level in the pyramid, the GSD increases by a factor of 2. The GSD at each level in the constructed pyramids is listed in Table 3.

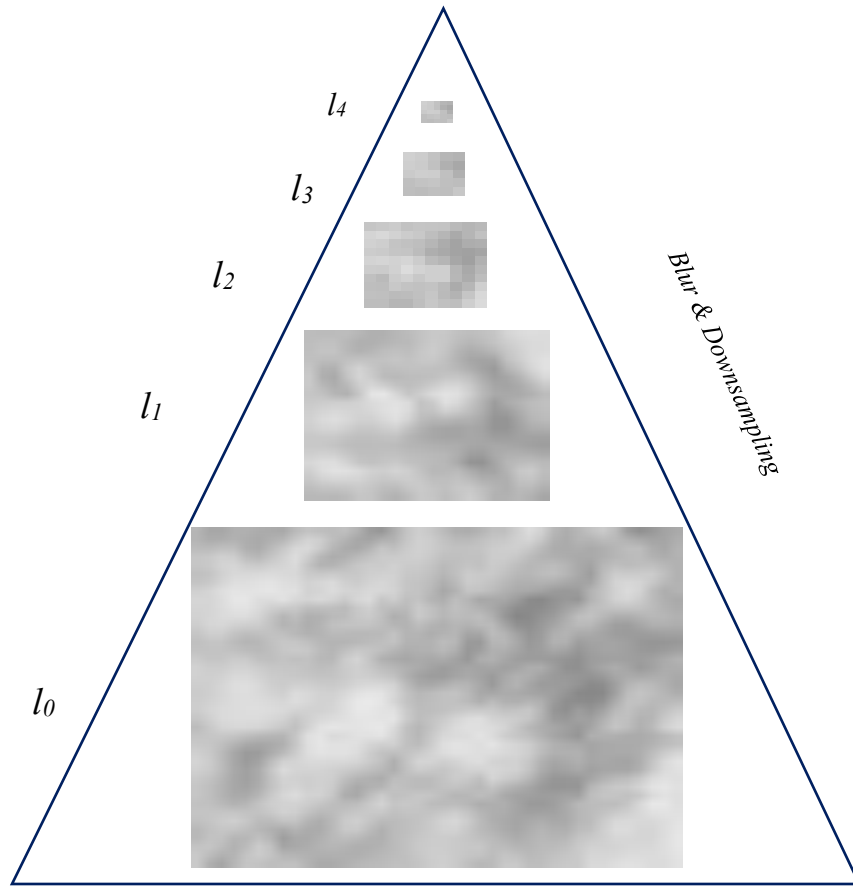


Figure 10: Gaussian Pyramid. The image illustrates five levels of the GP, spanning from the original image at level  $l_0$  to the fifth level,  $l_4$ . Note the extent of the physical domain represented by each image stays the same despite the reduction in the number of number of pixels constituting the image.

Table 3: Relation between each level of GP and GSD.

GP Level	Ground Sample Distance (GSD)
Level 1	$2.9\text{cm} \times 2 = 5.8 \text{ cm}$
Level 2	$5.8 \text{ cm} \times 2 = 11.6 \text{ cm}$
Level 3	$11.6 \text{ cm} \times 2 = 23.2 \text{ cm}$
Level 4	$23.2 \text{ cm} \times 2 = 46.4 \text{ cm}$

### *GLCM Features*

This work will use meta-pixel-based image analysis, as presented by Baron and Hill (2019) to identify the relative abundance of spotted knapweed in the study area. In this method, a chessboard segmentation is applied to divide each image into a set of non-overlapping squares, called metapixels, which are the same size as a field-survey quadrat (i.e., 1m<sup>2</sup>). Features to describe these metapixels are then calculated from the image pixels within the metapixel boundaries. In this work, 12 features were extracted from each metapixel are listed in Table 4 and include the mean and standard deviation of the pixel values and ten GLCM-based texture features. These 12 features were extracted for each metapixel in each of the 5 GPs, resulting in a set of 84 features describing metapixel per level in the GPs. Across all 5 levels in the GPs, this amounts to a set of 420 features per metapixel.

*Table 4: GLCM Extracted Features.*

<b>Extracted Features</b>
Mean of a pixel value
Standard deviation of a pixel value
Mean ASM
Range ASM
Mean Entropy
Range Entropy
Mean Sum Entropy
Range Sum Entropy
Mean Difference Entropy
Range Difference Entropy
Mean Correlation
Range Correlation



## Feature Compilation

Following the work of Baron and Hill (2020), the relative abundance of spotted knapweed within each surveyed quadrat was represented as a qualitative class rather than a quantitative proportion. This was done to ensure that there were sufficient examples in the dataset of each abundance of spotted knapweed to train a classifier. Three non-overlapping classes, “None”, “Moderate” and “High”, were defined. The definitions of these three classes, as well as the number of cases in the dataset of each class can be found in Table 5. For more information, see Baron et al. (2020) and Baron (2020) or section 1.8 of this thesis.

*Table 5: Classification of Spotted Knapweed Abundance in Surveyed Sites.*

<b>Abundance of Knapweed</b>	<b>Number of Cases</b>	<b>Qualitative Class</b>
Either absent or present in trace amounts	51	None
No more than 25% cover	63	Moderate
Exceeding 25%	67	High

The datasets for training and validating the classifiers in this study were generated by extracting 420 multi-scale features from metapixels delineated by the boundaries of 93 survey quadrats across three sites (31 quadrats were measured at each field site). These data were then split into two distinct subsets: a training set and a validation set. The validation set consisted solely of data collected on July 4, 2018, from the 31 quadrats in Site 3, while the training set comprised data from the remaining dates and sites (Site 1 on July 4th and 19th, Site 2 on July 4th and 12th, and Site 3 on July 12th and 19th). To enhance the performance of the classifiers stratified random sampling was used to construct a balanced training set (Zhu et al. 2016). A balanced training set has an equal number of examples representing each outcome classification. In this work, stratified random sampling was used to oversample the minority class and undersample the majority classes to equalize the sample sizes across all classes (Ma et al., 2015). For training both RF and SVM classifiers, stratified random sampling was used to create training sets that were then

used to train both the RF and SVM classifier. This approach ensured that each classifier was trained using the same set of training examples, allowing for a controlled comparison of their performance. The training set consisted of 150 samples in total: 50 in the 'High' category, 50 in the 'Moderate' category, and 50 in the 'None' category. Conversely, the validation set comprised 31 samples, with 17 in the 'High' category, 13 in the 'Moderate' category, and one representing the 'None' category.

## Model Creation

Classifiers based on Support Vector Machines (SVM) and Random Forests (RF) were developed to predict the relative abundance of spotted knapweed within a metapixel, utilizing the features specified in Table 4 calculated for each of the spectral band or vegetation index GPs. An SVM and RF classifier was developed using training data from each of the 5 levels in the data GPs. These classifiers are designated as GP0 through GP4, correlating with the pyramid levels from zero to four. Another pair of SVM and RF classifiers, hereafter indicated as 'GPs', was constructed using a feature set that concatenates the features across all the levels in the GPs.

Classifier construction began with feature selection to identify an optimal feature subset for classification. Recursive feature elimination driven by an RF classifier with 500 trees was used to identify the optimal feature set. Because of stochasticity in the RF training process, optimal feature selection was performed using a two-step ensemble method. The first step used recursive feature elimination to identify the optimal number of features ( $N_{opt}$ ). This process was repeated 20 times, and the average cross-validation score was tabulated for each possible number of features from 1 to N, where N is the entire feature set. The optimal number of features was identified as the number of features that generated the highest average cross-validation score. The second step used recursive feature elimination to identify the optimal feature set of size  $N_{opt}$ . This step was also repeated 20 times. These 20 feature sets were analyzed to find the most frequently selected features. The  $N_{opt}$  features that were selected most frequently in these 20 iterations were declared to be the optimal feature set.

Once the optimal feature subset was identified, these features were used along with a grid search to identify the optimal hyperparameter settings for the training process. The hyperparameters that were tuned, and the range of values considered by the grid search during this process are listed in Tables 8 and 9 for RF and SVM classifiers, respectively.

Finally, the model was trained using the optimal feature set and hyperparameters. This training was conducted via 10-fold cross-validation, and the best-performing model was retained as the final trained classifier. Each classifier was then tested using the independent validation data set consisting of data collected from site 3 on July 4. This approach ensures that the classifier's ability to generalize is accurately assessed using data that was not involved in the tuning or training processes. The analysis of the classifiers' performance is based on the performance metrics shown in Table 6 (Murphy, 2012).

*Table 6: Classifier performance metrics used to evaluate classifiers performance.*

<b>Classifier performance metrics</b>
Accuracy
Precision
Recall
F1-score
Macro Average
Weighted Average
Support

A flow chart of the classifier training and testing process is illustrated in Figure 11.

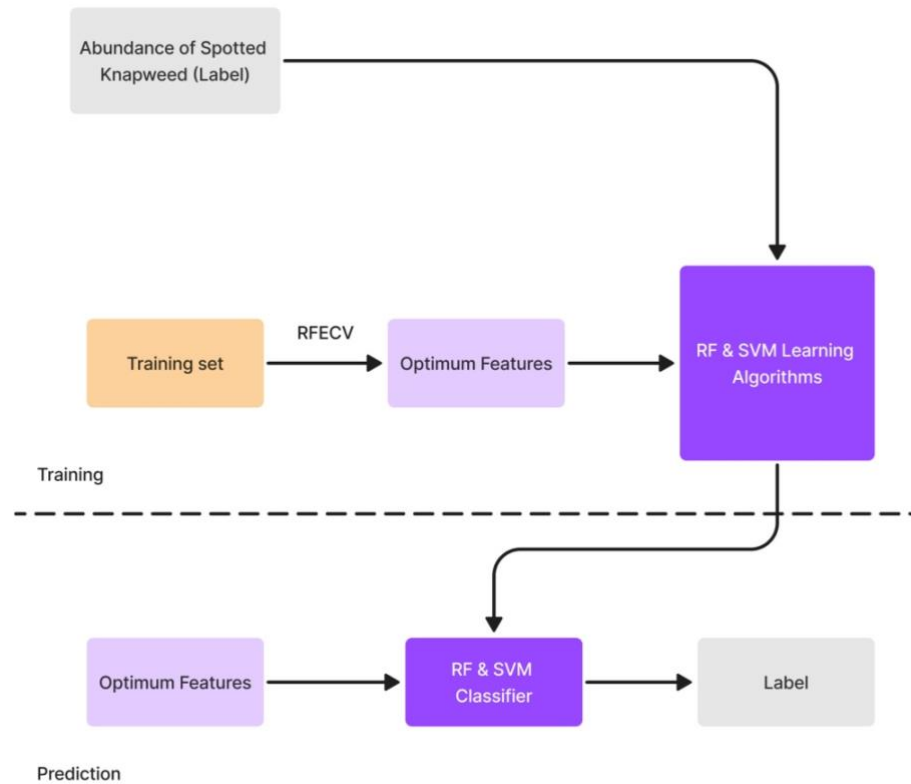


Figure 11: A procedure of classifying Spotted Knapweed using RF and SVM classifiers.

## Results

### Feature Selection

Model development began with identifying optimal features using the RFECV method, facilitated by a RF classifier composed of 500 trees. Figures 12 to 17 show the average cross-validation score versus number of features for each of the 6 feature sets (GP0, GP1, GP2, GP3, GP4, and GPs) the results of the RFECV process. In these plots, each point indicates the average cross-validation score over the 20 iterations of the RFECV process. These data are used to select the optimal number of features in each feature set by identifying the feature set size that optimizes the average cross-validation score.

Figure 12 illustrates the RFECV results for the GP0 data. The curve indicates the relationship between the number of features and the model's mean CV score. Starting from a low number of features, there is a notable rise in the CV score, which rapidly

increases and then levels off as more features are added. The peak of this curve is highlighted at the point '21, 0.534', where the model achieves the best performance with 21 features. Beyond this peak, the CV score tends to plateau, showing a consistent trend with minor fluctuations around this peak value. This plateau suggests that adding more than 21 features does not significantly improve the model's predictive capability and that the model has reached a balance between feature complexity and performance. The shape of the curve is characteristic of an initial gain in performance with additional features, followed by stabilization, indicating that further feature additions are not contributing to the predictive strength of the model.

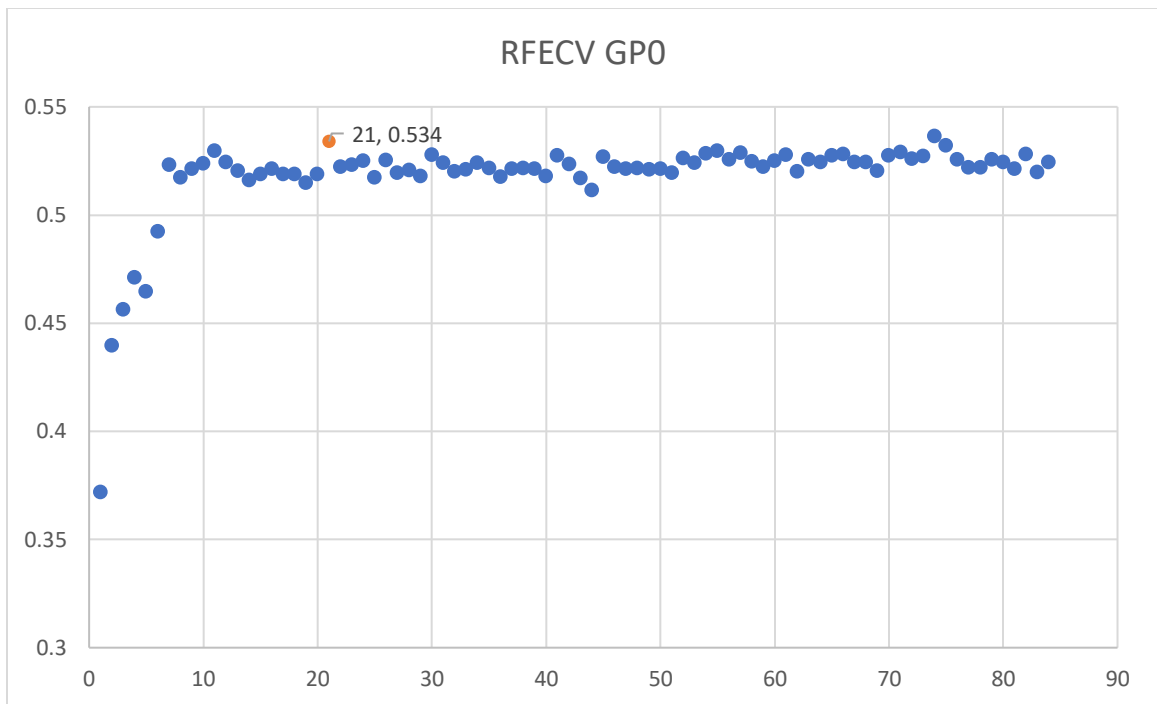


Figure 12: RFECV Results for GPO, showing the optimal selection of 21 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations.

Figure 13 illustrates the RFECV results for the GP1 data. The curve begins with a sharp increase, where the mean CV score swiftly rises from the lower number of features, reaching a prominent peak at '16, 0.567'. This suggests that at 16 features, the model

attains its highest level of predictive accuracy. Beyond this point, the performance sharply declines, demonstrating that additional features detract from the model's effectiveness. Following this decline, the curve levels off, signifying that further inclusion of features fails to significantly enhance the mean CV score. This leveling persists throughout the rest of the plot, marked by minor fluctuations but lacking any substantial upward movement. Such a pattern suggests that expanding the feature set beyond the optimal count of 16 leads to diminished performance. The curve's trajectory is indicative of a typical phenomenon in feature selection, whereby the marginal gain of adding extra features eventually plateaus or even reverses, reflecting the trade-off between model complexity and generalization.

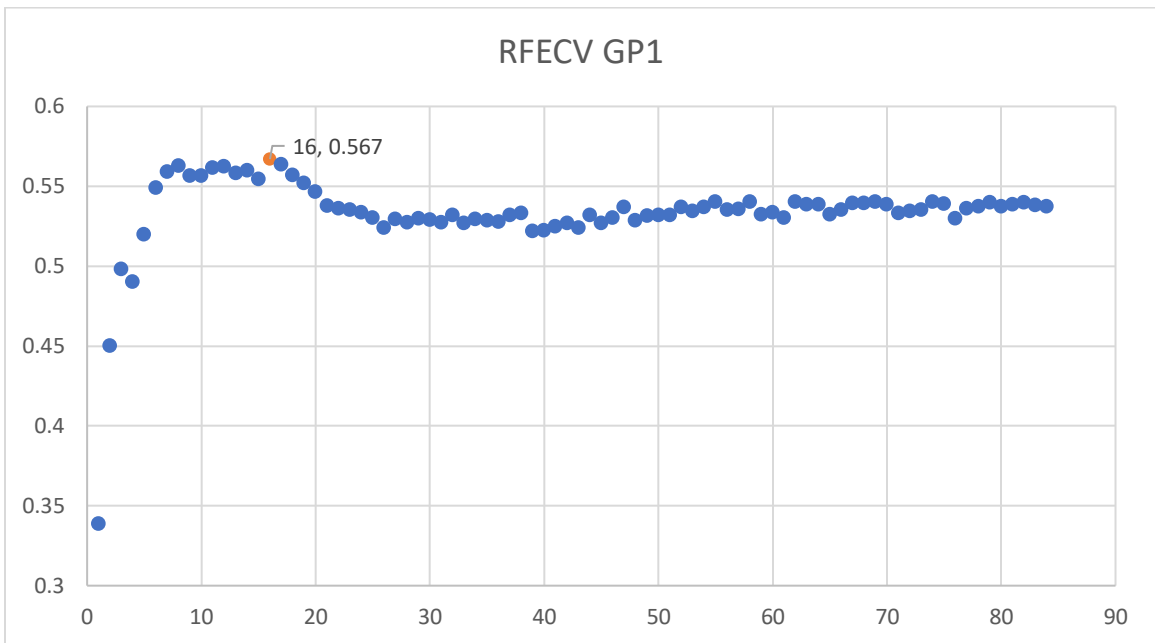


Figure 13: RFECV Results for GP1, showing the optimal selection of 16 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations.

Figure 14 illustrates the RFECV results for the GP2 data. The curve starts with a steep ascent in the mean CV score, reaching an early peak at the point '4, 0.582'. This indicates that the model's predictive accuracy is maximized with just 4 features. Following

this initial peak, the mean CV score decreases before plateauing, suggesting that additional features beyond the optimal four serve to decrease the model performance. The plateau is characterized by a consistent average score with minor fluctuations around the peak value, reinforcing the notion that a small, concise set of features is sufficient for the model to achieve its best performance. The profile of the curve demonstrates the principle of parsimony in model building, where simpler models with fewer features may yield the best generalization. It also illustrates the concept of diminishing returns in feature inclusion: beyond the optimal number, additional features do not contribute to improved model accuracy and may instead introduce noise or unnecessary complexity.

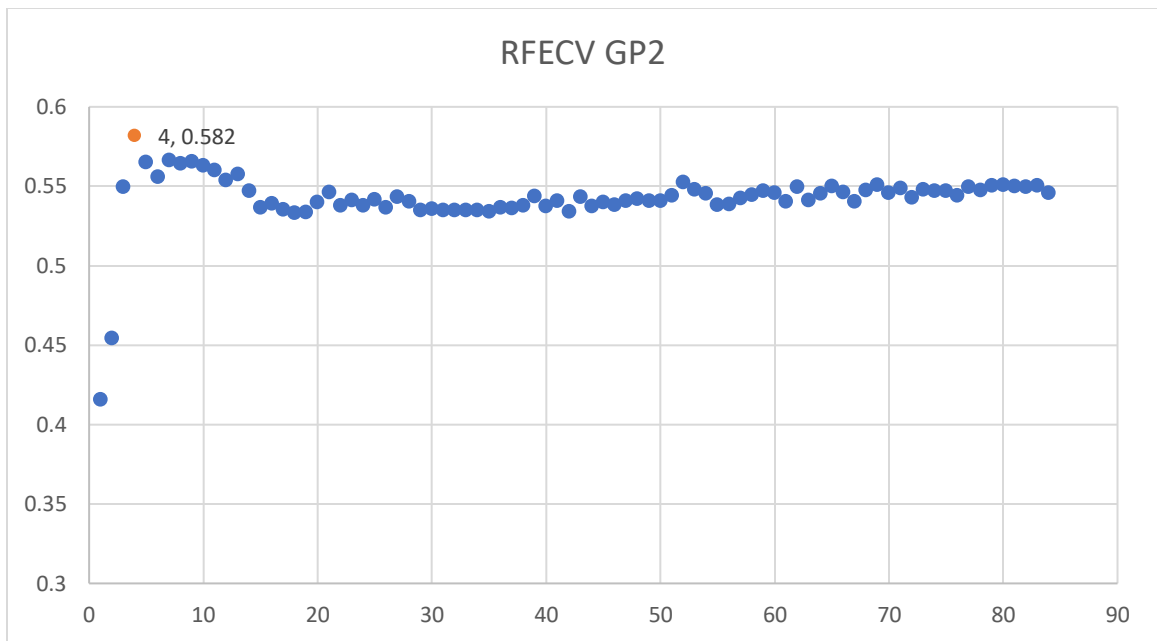


Figure 14: RFECV Results for GP2, showing the optimal selection of 4 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations.

Figure 15 illustrates the RFECV curve for the GP3 data. The mean CV score rises rapidly as more features are introduced into the model, indicating that each new feature contributes significantly to improving the model's predictive accuracy at this stage. The curve shows that the model is initially gaining valuable information from the incremental

addition of features, which is reflected in the increasing CV scores. This suggests that the features added early in the sequence are highly relevant and provide new, useful information that the model can leverage to improve its predictions. As we approach the peak at '16, 0.56', the rate of increase in the mean CV score begins to slow down, indicating that we are nearing the optimal number of features. Each additional feature contributes less to the model's performance, which is typical in feature selection processes where early additions have more impact, and the benefit of adding more features diminishes as the number of features grows. The peak itself represents the point of balance where the model has incorporated just enough features to maximize its performance without yet overfitting or incorporating redundant information. Beyond this peak, as we continue to add features, the performance will not improve further and even start to decline, as seen in the rest of the plot.

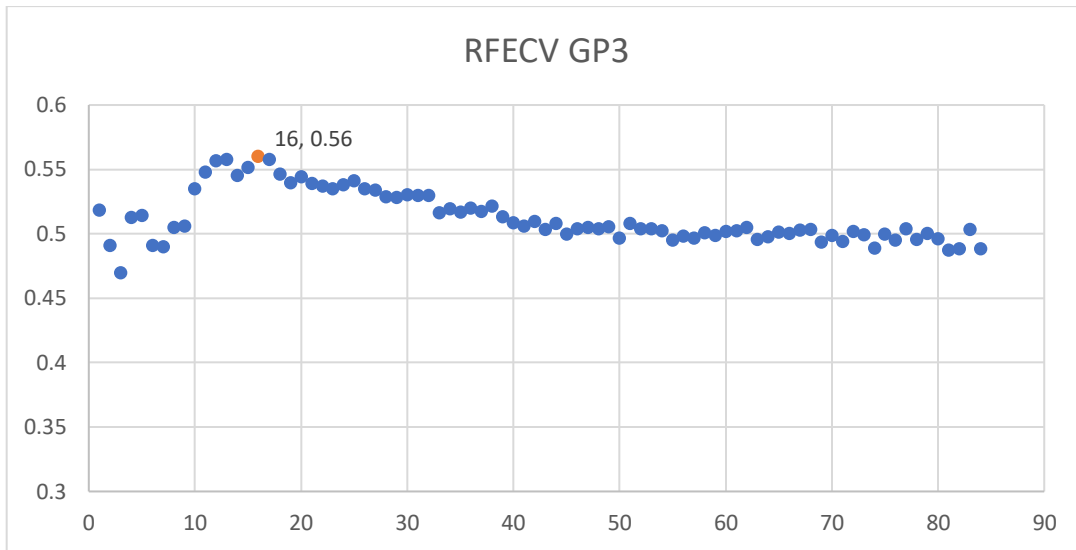


Figure 15: RFECV Results for GP3, showing the optimal selection of 16 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations.

Figure 16 illustrates the RFECV results for the GP4 data. In this figure, the mean cross-validation score ascends gradually, reflecting a steady improvement in model performance with each additional feature. This growth suggests that the early features added are each contributing meaningful information that the model is able to use to



enhance its accuracy. Beyond the peak at '8, 0.524333333', the curve levels off, indicating that the inclusion of more features does not lead to further significant gains in mean CV score. This plateau suggests that the model has captured the most informative features, and additional features do not contribute new information that could improve the model's performance. The relatively flat line continuing to the right of the peak implies that the additional features might be redundant or irrelevant, as they do not enhance the model's predictive power.

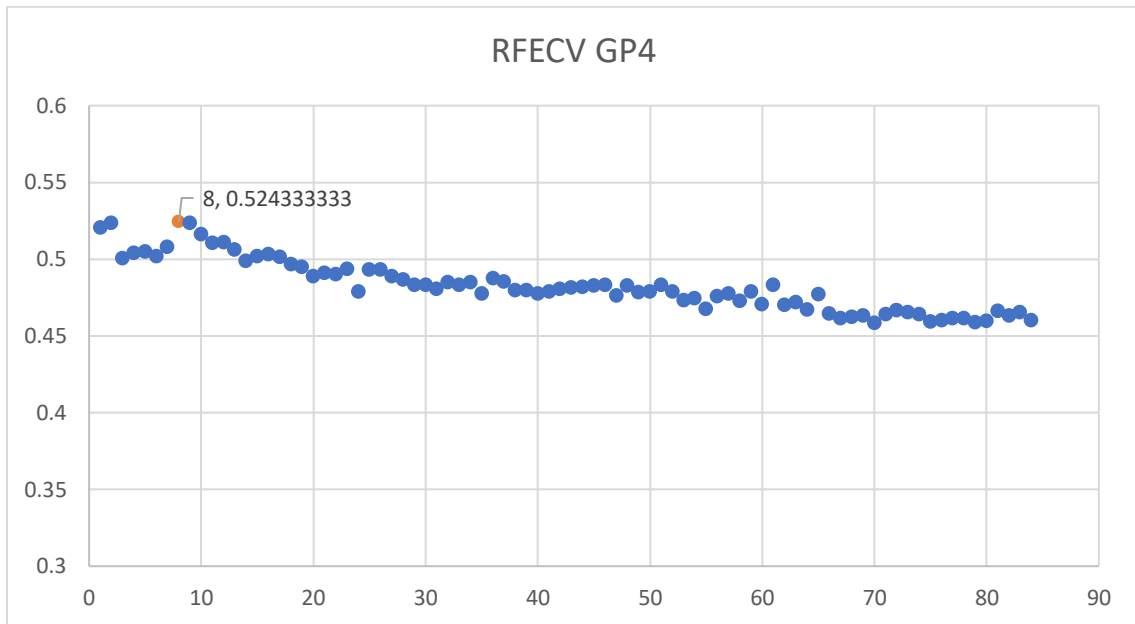


Figure 16: RFECV Results for GP4, showing the optimal selection of 8 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations.

In Figure 17, which visualizes the RFECV results for a combined feature set from all GP levels, the trajectory of the CV scores forms a distinctive shape. Initially, there's a rapid ascent in the mean CV scores as the number of features increases, reaching an apex at the point marked '5, 0.6304'. This peak signifies the most efficient number of features for the model, indicating that beyond this count, the additional features may not contribute significantly to the predictive power and might even introduce redundancy. After this

peak, the mean CV scores exhibit a gradual decline, suggesting a plateau effect followed by a slight downward trend as the number of features continues to grow. This pattern implies that the Random Forest algorithm, while generally robust to multicollinearity due to its feature bagging approach, is not completely resistant to the diminishing returns or potential adverse effects of including too many similar or non-informative features. The overall shape of the curve reinforces the principle that beyond a certain point, adding more features can lead to overfitting, where the model becomes overly complex and less generalizable to new data.

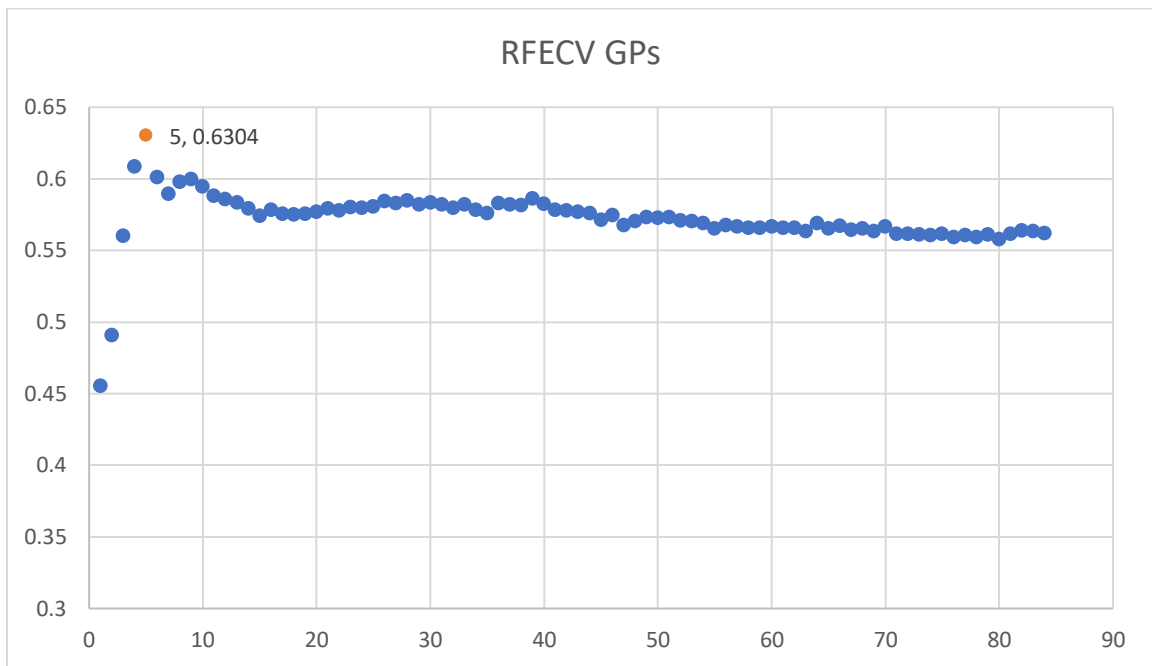


Figure 17: RFECV Results for GPs, showing the optimal selection of 5 features. The x-axis represents the number of features retained, and the y-axis depicts the average of the mean cross-validation scores, as calculated over 20 iterations.

Once the size of the optimal feature set was identified for each data set, another iterative process was conducted to select the features that constituted the optimal feature set for each data set. This process involved iteratively performing recursive feature elimination with an RF model, consisting of 500 trees, to identify a data set of size  $n_{opt}$ , where  $n_{opt}$  is the size of the feature set identified in Figures 12 through 17. This feature

selection step was repeated 20 times, and the features selected in each iteration were counted. The  $n_{opt}$  features which were selected the most often during this process were declared to be the optimal feature set for the input data.

During this feature selection process, 40 out of 84 features were included in the optimal feature set for at least one of the GP levels considered. In Table 7 the color-coding helps to quickly visualize which features are most frequently selected across different GP levels, indicating their importance and relevance in the classification task. Features that appear in more columns are likely to be more robust for the model. The 'nir\_mean' and 'rededge\_mean' features stand out as they are consistently chosen across all GP levels, suggesting that they are significant predictors regardless of the GP level or when using a combination of all levels.

Table 7: Optimized Features for GP0, GP1, GP2, GP3, GP4 and concatenated GPs.

Feature Names	GP0	GP1	GP2	GP3	GP4	GPs	Counts
nir_mean							6
rededge_mean							6
reNDVI_mean							5
green_sd							4
green_mean							3
green_meanEntropy							3
green_rangeASM							3
reNDVI_meanCorrelation							2
NDVI_meanEntropy							2
red_rangeEntropy							2
green_meanASM							2
green_meanSumEntropy							2
green_meanDifferenceEntropy							2
gNDVI_mean							2
reNDVI_rangeSumEntropy							1
NDVI_meanASM							1
NDVI_meanSumEntropy							1
NDVI_meanDifferenceEntropy							1
NDVI_rangeASM							1
NDVI_rangeSumEntropy							1
nir_meanASM							1
nir_rangeASM							1
nir_rangeCorrelation							1
nir_rangeSumEntropy							1
rededge_sd							1
rededge_meanASM							1
rededge_rangeCorrelation							1
rededge_rangeEntropy							1
red_sd							1
red_meanASM							1
red_meanSumEntropy							1
red_meanDifferenceEntropy							1
red_rangeCorrelation							1
gNDVI_sd							1
gNDVI_meanASM							1
gNDVI_meanSumEntropy							1
gNDVI_meanEntropy							1
gNDVI_rangeASM							1
gNDVI_rangeSumEntropy							1
gNDVI_rangeDifferenceEntropy							1

## Hyperparameter Optimization

A grid search was used to identify the optimal hyperparameters for each classifier developed in this work. The hyperparameters tuned and the range of values included in the grid search are listed in Table 8 and 9 for the RF and SVM models, respectively.

*Table 8: Range of hyperparameter values considered for tuning the Random Forest Classifier using Grid Search.*

bootstrap	TRUE, False
max_depth	20, 40, 80, 100
max_features	sqrt
min_samples_leaf	Integers (1,10)
min_samples_split	Integers (2,11)
n_estimators	400, 500, 600, 800, 1000, 1200,1500

*Table 9: Range of hyperparameter values considered for tuning SVM using Grid Search.*

C	0.1, 1, 10, 100
Degree	2, 3, 4, 5
Gamma	Scale, Auto, 0.1, 1, 10
Kernel	linear, Rbf, Poly, Sigmoid

Table 10 lists the optimized hyperparameters identified for the RF models developed using data from each of the 5 levels of the GP (GP0 to GP4) and the composite of all levels (GPs). The table is organized with hyperparameters listed in rows and the GP levels, including the combined GPs, in columns. The 'bootstrap' hyperparameter indicates whether bootstrap sampling is used when building trees; it is set to FALSE for GP0, GP1, GP2, and the combined GPs; thus, the entire training set is utilized for tree construction. Conversely, for GP3 and GP4, this parameter is TRUE, signifying the use of bootstrap

samples. The 'max\_depth' hyperparameter, which restricts tree depth to prevent overfitting, was set at 20 for all levels. The 'Max\_features' hyperparameter determines the number of features considered at each split. The square-root of the number of features was constantly selected across all levels. Variability is observed in 'min\_samples\_leaf' and the 'min\_samples\_split' hyperparameters. The 'min\_samples\_leaf' parameter sets the minimum samples at a leaf node, while 'min\_samples\_split' parameter specifies the threshold for splitting nodes. The 'n\_estimators' hyperparameter, reflecting the number of trees in the forest, also takes values for the data aggregated at different levels of the GPs. This suggests that the pattern complexity may change at different levels of spatial aggregation. Because the GPs data set has features at different levels of spatial aggregation, it is expected that these data require a larger number of trees for optimal performance.

Post-tuning, an elevation in cross-validation scores signifies that the model's accuracy has been enhanced, demonstrating the importance of hyperparameter tuning for building effective RF classifiers.

*Table 10: Result of RF hyperparameters tuning for GP0 to GP4 and GPs.*

Hyperparameters	GP0	GP1	GP2	GP3	GP4	GPs
bootstrap	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE
max_depth	20	20	20	20	20	20
max_features	sqrt	Sqrt	sqrt	sqrt	sqrt	sqrt
min_samples_leaf	1	1	1	2	3	1
min_samples_split	4	4	15	2	11	5
n_estimators	500	400	500	600	400	800

Table 11 lists the optimized values of the SVM hyperparameters for the data at different GP levels and the concatenated dataset encompassing all levels (GPs). A notable variation in the 'C' parameter across the GP levels indicates that unique regularization strengths are optimal at each level, tailored during the hyperparameter tuning phase to enhance model performance. The 'Gamma' parameter, set to 'Scale' for GP0, GP4, and the combined GPs, is adjusted automatically relative to the dataset's feature count, suggesting an adaptive approach to feature influence. In contrast, fixed values are assigned to 'Gamma' for GP1, GP2, and GP3, denoting a precise calibration for these individual levels. The 'Kernel' choice remains consistent with 'RBF' for GP0 through GP4, indicating a preference for this kernel's ability to handle non-linear relationships. Conversely, a 'Linear' kernel is chosen for the combined GPs, suggesting an underlying linear separability when the GP levels are merged.

*Table 11: Result of SVM hyperparameters tuning for GP0 to GP4 and GPs.*

Hyperparameters	GP0	GP1	GP2	GP3	GP4	GPs
C	100	1	100	10	1	10
Degree	2	2	2	2	2	2
Gamma	Scale	0.1	0.1	0.1	Scale	Scale
Kernel	RBF	RBF	RBF	RBF	RBF	Linear

## Classifier Performance

### *Classification Based on the GP0 Feature Set*

Figures 18 and 19 illustrate the respective performance of the GP0 RF and SVM classifiers on the validation dataset. Each cell of the confusion matrix shows the proportion of the total number of predictions that fall into the corresponding category. For instance, in Figure 18, the cell in the first row and first column indicates that 82% of the 'High' class was correctly predicted. The middle cell in the first row of this figure shows that 6% of the 'High' class instances were incorrectly predicted as 'None', and the cell in the first row and third column shows that 12% of the 'High' class instances were

incorrectly predicted as 'Moderate'. The color gradient, ranging from light to dark green, reflects the magnitude of the proportions, with darker shades representing higher proportions.

		Predicted Label		
		High	Moderate	None
True Label	High	14	1	2
	Moderate	5	2	6
	None	0	0	1

*Figure 18: RF confusion matrix for the GPO validation set.*

		Predicted Label		
		High	Moderate	None
True Label	High	10	5	2
	Moderate	5	1	7
	None	1	0	0

*Figure 19: SVM confusion matrix for GPO validation set.*

Table 12 reports the performance of both the RF and SVM GPO classifiers on the validation data set from July 4.

Table 12: RF and SVM classification results for the GP0 feature set.

	Precision RF	Precision SVM	Recall RF	Recall SVM	F1- score RF	F1- score SVM	Support RF	Support SVM
High	0.82	0.62	0.74	0.59	0.78	0.61	19	17
None	0	0	0	0	0	0	3	1
Moderate	0.46	0.78	0.67	0.54	0.55	0.64	9	13
Macro Avg	0.43	0.47	0.47	0.38	0.44	0.41	31	31
Weighted Avg	0.64	0.67	0.65	0.55	0.64	0.6	31	31
Accuracy				Value		Support		
RF				0.65		31		
SVM				0.55		31		

From this table, it can be seen that:

- High Class:** For the High class, the RF classifier outperforms the SVM in both precision (0.82 vs. 0.62) and recall (0.74 vs. 0.59), indicating that RF is more accurate and also more comprehensive in identifying true High-class cases. The F1-score, which balances precision and recall, is correspondingly higher for RF (0.78 vs. 0.61), affirming its superior performance in this class. This suggests that RF is more adept at handling instances in this category, which could be attributed to its ensemble approach, potentially capturing more complex patterns within the High-class features than the SVM.



- **Moderate Class:** In the Moderate class, SVM shows a higher precision (0.78 vs. 0.46) but lower recall (0.54 vs. 0.67) compared to RF. While SVM is better at correctly labeling Moderate-class instances when it does predict them, it fails to identify a significant proportion of actual Moderate-class cases, as evidenced by the lower recall. RF, although less precise, is more reliable in identifying the presence of the Moderate class, but it also misclassifies more non-Moderate instances as Moderate. The F1-scores are quite close (0.55 for RF and 0.64 for SVM), suggesting a trade-off between precision and recall for the two models.
- **None Class:** Both classifiers fail to identify any instances of the None class, with all scores at 0. This indicates a significant challenge for both models in detecting this class, which could be due to an extremely small number of instances (Support for RF is 3 and SVM is 1). The lack of learning material for this class makes it difficult for both classifiers to establish a pattern, rendering them ineffective for the None-class.

When comparing the overall performance, RF demonstrates a higher accuracy (0.65) than SVM (0.55). This suggests that across all classes, RF maintains a better balance between precision and recall, leading to more accurate classification results. The Macro Avg and Weighted Avg scores also support this, with RF showing a slight edge in performance over SVM. These metrics suggest that RF is generally more effective in classifying instances across this dataset, especially considering the Weighted Avg, which takes the support of each class into account, reflecting the real-world distribution of classes.

In summary, RF tends to have a more balanced performance across the High and Moderate classes, while SVM struggles with recall but has instances where it can be highly precise. Neither model performs well in the None class. This is probably due to an insufficient number of examples of this class in the training data. Overall, RF is more accurate and consistent across classes, making it a more reliable choice for classification in this scenario.

### *Classification Based on the GP1 Feature Set*

Figures 20 and 21 illustrate the respective performance of the GP1 RF and SVM classifiers on the validation dataset.

		Predicted Label		
		High	Moderate	None
True Label	High	11	1	5
	Moderate	4	1	8
	None	0	0	1

*Figure 20: RF confusion matrix for the GP1 validation set.*

		Predicted Label		
		High	Moderate	None
True Label	High	12	2	3
	Moderate	7	2	4
	None	0	1	0

*Figure 21: SVM confusion matrix for the GP1 validation set.*

Table 13 reports the performance of both the RF and SVM classifiers on the GP1 validation data set from July 4.

Table 13: RF and SVM classification results for the GP1 feature set.

	Precision RF	Precision SVM	Recall RF	Recall SVM	F1- score RF	F1- score SVM	Support RF	Support SVM
High	0.65	0.63	0.73	0.71	0.69	0.67	15	17
None	0	0.2	0	1	0	0.33	2	1
Moderate	0.62	0.57	0.57	0.31	0.59	0.4	14	13
Macro Avg	0.42	0.47	0.43	0.67	0.43	0.47	31	31
Weighted Avg	0.59	0.59	0.61	0.55	0.6	0.54	31	31
Accuracy				Value		Support		
RF				0.61		31		
SVM				0.55		31		

From this table, it can be seen that:

- High Class:** Both classifiers perform comparably in precision for the High class, with RF at 0.65 and SVM at 0.63. However, RF has a slightly better recall of 0.73 compared to SVM's 0.71, indicating that RF is marginally better at capturing the majority of High-class cases. The F1-scores are also similar, with RF at 0.69 and SVM at 0.67, indicating balanced precision and recall for both classifiers in this category.
- Moderate Class:** In the Moderate class, RF demonstrates a higher precision (0.62 vs. 0.57) and recall (0.57 vs. 0.31) compared to SVM, which is mirrored in the F1-score (0.59 for RF vs. 0.4 for SVM). This indicates that RF is more adept at correctly

identifying and capturing Moderate-class instances than SVM, which is less reliable in recognizing true Moderate cases.

- **None Class:** The None class presents an interesting contrast. RF fails to identify any None-class instances (precision, recall, and F1-score at 0), whereas SVM, despite its limited precision at 0.2 and perfect recall at 1, manages an F1-score of 0.33. This suggests that while SVM is able to recognize the None-class instances, it tends to misclassify other classes as None, as evidenced by its low precision.

Based on these results, it can be seen that RF is more consistent and accurate across the majority of classes and the dataset as a whole. SVM shows some strengths, particularly in the None class, but its higher Macro Average recall does not translate to better overall accuracy or balance between precision and recall, as reflected in the lower overall accuracy and Weighted Average F1-score. RF's higher values in these key metrics make it the more reliable classifier for this particular dataset at GP1.

### *Classification Based on the GP2 Feature Set*

Figures 22 and 23 illustrate the respective performance of the GP2 RF and SVM classifiers on the validation dataset.

		Predicted Label		
		High	Moderate	None
True Label	High	10	1	6
	Moderate	1	2	10
	None	0	1	0

*Figure 22: RF confusion matrix for the GP2 validation set.*

		Predicted Label		
		High	Moderate	None
True Label	High	16	1	0
	Moderate	8	0	5
	None	1	0	0

*Figure 23: SVM confusion matrix for the GP2 validation set.*

Table 14 reports the performance of both the RF and SVM classifiers on the validation data set from July 4.

Table 14: RF and SVM classification results for the GP2 feature set.

	Precision RF	Precision SVM	Recall RF	Recall SVM	F1- score RF	F1- score SVM	Support RF	Support SVM
High	0.59	0.64	0.91	0.94	0.71	0.76	11	17
None	1	0	0.25	0	0.4	0	4	1
Moderate	0.77	1	0.62	0.38	0.69	0.56	16	13
Macro Avg	0.79	0.55	0.59	0.44	0.6	0.44	31	31
Weighted Avg	0.73	0.77	0.66	0.68	0.66	0.65	31	31
Accuracy				Value		Support		
RF				0.68		31		
SVM				0.68		31		

From this table, it can be seen that:

- **High Class:** For the High class, SVM has a slightly higher precision (0.64) than RF (0.59), suggesting that SVM is marginally better at correctly identifying the High class when it predicts an instance as High. In recall, SVM also has an edge (0.94) over RF (0.91), indicating that SVM is better at capturing the true High-class instances within the dataset. The F1-score, which considers both precision and recall, is higher for

SVM (0.76) compared to RF (0.71), confirming SVM's better performance for the High class.

- **Moderate Class:** SVM shows perfect precision (1.0) for the Moderate class, which means all instances SVM predicts as Moderate is correct. However, its recall is only 0.38, indicating it misses a large number of true Moderate-class instances. RF has a lower precision (0.77) but a higher recall (0.62), which suggests it captures more of the Moderate-class instances and has some false positives. The F1-score is higher for RF (0.69) than SVM (0.56), indicating a better balance of precision and recall for RF in the Moderate class.
- **None Class:** RF achieves perfect precision (1.0) but has a low recall (0.25), which means it can correctly identify the None class when it predicts it, but it misses many actual instances of the None class. SVM, on the other hand, has zero recall, indicating it failed to identify any true None-class instances, leading to an F1-score of 0. Despite RF's limited recall, its ability to identify some Low-class instances gives it a better F1-score (0.4) compared to SVM (0), which completely fails in this class.

The Macro Average precision is significantly higher for RF (0.79) compared to SVM (0.55), suggesting RF is more precise on average across all classes. The Macro Average recall is also better for RF (0.59 vs. 0.44 for SVM), indicating RF is more effective at capturing true positives across the board. The Weighted Average precision is slightly better for SVM (0.77) compared to RF (0.73), while the Weighted Average recall is the same for both classifiers (0.68), leading to very similar F1-scores (RF: 0.66, SVM: 0.65). The overall accuracy for both RF and SVM is the same (0.68), indicating that both classifiers correctly predict the class labels for 68% of the dataset.

While RF and SVM have the same overall accuracy at the GP2 level, RF performs better in the Moderate and None classes, particularly in terms of recall. SVM, however, performs slightly better in the High class, with better precision and recall. The Macro Averages favor RF, indicating that it has a better average performance across all classes,

but the Weighted Averages are very similar, reflecting the balanced performance of both classifiers when the class distribution is taken into account.

### *Classification Based on the GP3 Feature Set*

Figures 24 and 25 illustrate the respective performance of the GP3 RF and SVM classifiers on the validation dataset.

		Predicted Label		
		High	Moderate	None
True Label	High	12	3	2
	Moderate	6	1	6
	None	0	1	0

*Figure 24: RF confusion matrix for the GP3 validation set.*

		Predicted Label		
		High	Moderate	None
True Label	High	12	2	3
	Moderate	6	1	6
	None	1	0	0

*Figure 25: SVM confusion matrix for the GP3 validation set.*

Table 15 reports the performance of both the RF and SVM classifiers on the validation data set from July 4.

Table 15: RF and SVM classification results for the GP3 feature set.

	Precision RF	Precision SVM	Recall RF	Recall SVM	F1- score RF	F1- score SVM	Support RF	Support SVM
High	0.71	0.63	0.67	0.71	0.69	0.67	18	17
None	1	0	0.2	0	0.33	0	5	1
Moderate	0.46	0.67	0.75	0.46	0.57	0.55	8	13
Macro Avg	0.72	0.43	0.54	0.39	0.53	0.4	31	31
Weighted Avg	0.69	0.63	0.61	0.58	0.6	0.59	31	31
Accuracy			Value			Support		
RF			0.61			31		
SVM			0.58			31		

From this table, it can be seen that:

- **High Class:** RF shows a precision of 0.71, higher than SVM's 0.63. This suggests that when RF predicts an instance as High, it is more likely to be correct. However, SVM has a slightly higher recall (0.71) than RF (0.67), indicating SVM is marginally better at identifying all relevant instances of the High class within the dataset. The F1-scores for both classifiers are nearly identical (RF: 0.69, SVM: 0.67), suggesting that both classifiers have a similar balance between precision and recall for the High class.
- **Moderate Class:** SVM shows higher precision (0.67) compared to RF (0.46), suggesting SVM is more accurate when it labels an instance as Moderate. Conversely, RF demonstrates a higher recall (0.75) over SVM (0.46), which suggests that RF is better



at detecting the true Moderate-class instances but also has more false positives. RF has an F1-score of 0.57, slightly higher than SVM's 0.55, indicating a slightly better balance between precision and recall for RF in the Moderate class.

- **None Class:** RF achieves a perfect precision (1.0) but has a very low recall (0.2), indicating it is selective and accurate when predicting an instance as None but failing to identify most actual None instances. SVM does not correctly identify any None-class instances, as evidenced by a recall of 0, leading to an F1-score of 0, which indicates a complete miss for the None class by the SVM. Despite its low recall, RF's ability to identify some None-class instances results in an F1-score of 0.33, indicating a better performance than SVM for the None class.

The Macro Average precision is substantially higher for RF (0.72) than SVM (0.43), indicating that RF is, on average, more precise across all classes. The Macro Average recall for RF (0.54) is also higher than SVM (0.39), suggesting that RF captures true positives across all classes more effectively. The Weighted Average precision for RF (0.69) is greater than SVM's (0.63), and the Weighted Average recall for RF (0.61) is also higher than SVM's (0.58). This results in a slightly better F1-score for RF (0.6 vs. 0.59 for SVM), again showing a more balanced performance. The overall accuracy is higher for RF (0.61) than SVM (0.58), indicating that RF is more effective across the entire dataset.

In summary, RF generally exhibits a better performance across the High and Moderate classes and a significantly better performance for the None class despite its low recall. RF's higher Macro and Weighted Average metrics, as well as its higher overall accuracy, indicate that it is the more reliable classifier for this dataset at the GP3 level. SVM may have its strengths, particularly in precision within the Moderate class, but RF provides a more consistent and effective performance overall.

### *Classification Based on the GP4 Feature Set*

Figures 26 and 27 illustrate the respective performance of the GP4 RF and SVM classifiers on the validation dataset.

		Predicted Label		
		High	Moderate	None
True Label	High	10	2	5
	Moderate	7	1	5
	None	0	1	0

*Figure 26: RF confusion matrix for the GP4 validation set.*

		Predicted Label		
		High	Moderate	None
True Label	High	15	1	1
	Moderate	9	0	4
	None	1	0	0

*Figure 27: SVM confusion matrix for the GP4 validation set.*

Table 16 reports the performance of both the RF and SVM classifiers on the validation data set from July 4.

Table 16: RF and SVM classification results for the GP4 feature set.

	Precision RF	Precision SVM	Recall RF	Recall SVM	F1- score RF	F1- score SVM	Support RF	Support SVM
High	0.59	0.6	0.59	0.88	0.59	0.71	17	17
None	1	0	0.25	0	0.4	0	4	1
Moderate	0.38	0.8	0.5	0.31	0.43	0.44	10	13
Macro Avg	0.66	0.47	0.45	0.4	0.47	0.39	31	31
Weighted Avg	0.58	0.66	0.52	0.61	0.51	0.58	31	31
Accuracy				Value		Support		
RF				0.52		31		
SVM				0.61		31		

From this table, it can be seen that:

- **High Class:** The precision for RF is slightly lower (0.59) than for SVM (0.6), which means SVM is marginally more accurate when it predicts an instance as High. However, SVM significantly outperforms RF in recall (0.88 vs. 0.59), suggesting that SVM is much better at identifying all relevant High-class instances. Consequently, the F1-score for SVM (0.71) is notably higher than for RF (0.59), indicating a better balance between precision and recall for SVM in the High class.
- **Moderate Class:** SVM exhibits a much higher precision (0.8) compared to RF (0.38), which means it is more accurate when it labels an instance as Moderate. RF has a higher recall (0.5) over SVM (0.31), suggesting that RF is better at detecting true Moderate-class instances but also includes more false positives. The F1-scores are

similar but slightly higher for SVM (0.44) than for RF (0.43), indicating a slightly better balance for SVM in the Moderate class.

- **None Class:** RF achieves perfect precision (1.0), indicating it correctly identifies all instances it labels as None, but its recall is very low (0.25), which means it misses most actual None instances. SVM does not identify any true None-class instances, reflected by a recall of 0 and, consequently, an F1-score of 0. RF, therefore, performs better for the None class with a modest F1-score of 0.4 due to its ability to identify some true None instances.

The Macro Average precision is higher for RF (0.66) compared to SVM (0.47), indicating that RF is more precise across all classes on average. However, the Macro Average recall is slightly higher for SVM (0.4) compared to RF (0.45), suggesting that SVM is better at capturing true positives across all classes. The Weighted Average precision is higher for SVM (0.66) compared to RF (0.58), while the Weighted Average recall is higher for SVM (0.61) compared to RF (0.52), leading to a slightly better F1-score for SVM (0.58 vs. 0.51 for RF). Notably, the overall accuracy is higher for SVM (0.61) than for RF (0.52), indicating that SVM is more effective at correctly predicting class labels across the entire dataset.

For GP4, while RF demonstrates higher precision on average across all classes, SVM has a better recall for the High class and a higher overall accuracy. This suggests that SVM is more adept at classifying this dataset, particularly for the most represented High class, which seems to drive its higher overall performance. Despite RF's perfect precision in the None class, its failure to identify the majority of None instances results in its lower overall accuracy.

### *Classification Based on the Concatenated GPs Feature Set*

Figures 28 and 29 illustrate the respective performance of the GPs RF and SVM classifiers on the validation dataset.

		Predicted Label		
		High	Moderate	None
True Label	High	4	5	8
	Moderate	6	0	7
	None	0	0	1

*Figure 28: RF confusion matrix for the concatenated GPs validation set.*

		Predicted Label		
		High	Moderate	None
True Label	High	10	4	3
	Moderate	10	1	2
	None	1	0	0

*Figure 29: SVM confusion matrix for the concatenated GPs validation set.*

Table 17 reports the performance of both the RF and SVM classifiers on the validation data set from July 4.

Table 17: RF and SVM classification result for the concatenated GPs feature set.

	Precision RF	Precision SVM	Recall RF	Recall SVM	F1- score RF	F1- score SVM	Support RF	Support SVM
High	0.24	0.5	0.41	0.6	0.3	0.54	49	85
None	0	0.08	0	0.4	0	0.13	23	5
Moderate	0.55	0.42	0.43	0.17	0.49	0.24	83	65
Macro Avg	0.26	0.33	0.28	0.39	0.26	0.3	155	155
Weighted Avg	0.37	0.45	0.36	0.41	0.35	0.4	155	155
Accuracy				Value		Support		
RF				0.36		155		
SVM				0.41		155		

From this table, it can be seen that:

- **High Class:** RF has low precision (0.24) and moderate recall (0.41) for the High class, resulting in a low F1-score (0.3). This indicates that RF isn't very accurate when it identifies an instance as High and misses a significant number of High instances. SVM outperforms RF in both precision (0.5) and recall (0.6) for the High class, with a substantially higher F1-score (0.54). This suggests that SVM is not only more accurate when it predicts an instance as High but also better at capturing more of the true High instances.
- **Moderate Class:** RF shows decent precision (0.55) and moderate recall (0.43) for the Moderate class, with a corresponding F1-score (0.49). This indicates that RF is

relatively accurate and reliable in identifying Moderate-class instances. SVM has lower precision (0.42) and significantly lower recall (0.17) for the Moderate-class compared to RF, resulting in a lower F1-score (0.24). This suggests SVM is less effective at detecting true Moderate instances.

- **None Class:** RF fails to identify any None-class instances, with precision, recall, and F1-score all at 0. This indicates a significant limitation of the RF model in detecting the None class. SVM shows very low precision (0.08) but a higher recall (0.4) for the None class, leading to a low F1-score (0.13). While SVM does manage to identify some true None-class instances, it also has a high rate of false positives.

The Macro Average precision is slightly higher for SVM (0.33) than RF (0.26), while the Macro Average recall is higher for SVM (0.39) compared to RF (0.28). This could indicate that SVM has a slight edge in detecting true positives on average across all classes. The Weighted Average precision is higher for SVM (0.45) than for RF (0.37), and the Weighted Average recall is also higher for SVM (0.41) compared to RF (0.36). This suggests that SVM has a better overall performance when considering the distribution of classes in the dataset. The overall accuracy is higher for SVM (0.41) than for RF (0.36), indicating that SVM is more effective at correctly classifying instances across the concatenated GP levels.

In conclusion, SVM generally shows better performance than RF across concatenated GP levels, particularly in the High class, which drives its higher overall accuracy. RF has its strengths, performing better in the Moderate class, but fails to identify any instances in the None class. Despite its limitations, SVM provides a more balanced performance across all classes, making it the preferable model in this comparative analysis.

## Discussion

Table 7 reveals significant insights into the feature selection process for image classification tasks. It shows that out of 84 features, less than half (40) are useful for classifying the relative abundance of spotted knapweed, which emphasizes the importance of feature optimization in image classification. The analysis of RF classifier performance as a function of the number of features used for classification, as shown in Figures 12 through 17, indicates that increasing the number of features initially benefits

model performance up to a certain point. Beyond this optimal number, however, including additional features tends to diminish the classifier's effectiveness, underscoring that even for models tolerant to multicollinearity, like RF, there's a threshold beyond which feature redundancy becomes counterproductive.

Of all the features considered in this study, the mean NIR and mean red-edge reflectance features appear to be the most useful for classifying the relative abundance of spotted knapweed, because they are included in the optimal feature sets for all GP levels and the concatenated GPs set. This selection aligns with our understanding of plant physiology; the chlorophyll absorption and cell structure reflection properties in these spectral ranges are critical for vegetation identification. The consistent selection of these features across different scale-space models underscores their robustness and importance in capturing the spectral signature of vegetation. Surprisingly, mean red reflectance, typically a vital component in traditional vegetation indices like NDVI, is absent from the optimized feature sets. This omission might indicate that the models leverage other features that capture similar information. For example, the reNDVI, which is formulated using NIR and red-edge bands, appears in all but the GP2 feature set, but the combination of these higher-wavelength spectra provides a significant capability for vegetation classification.

Green spectral band features, such as the mean and standard deviation of green reflectance and, the mean of the green-band GLCM-based entropy, and the range of the green-band GLCM-based angular second moment (ASM), appear in half of the optimized feature sets. Their presence highlights the role of spatial pattern of green band reflectance in capturing vegetation structure and texture, which may be pertinent to distinguishing spotted knapweed, especially considering its sparse canopy that allows for shadowing effects and visibility of the ground layer.

Aside from the NIR, red-edge, and certain green band features, most other features might be specific to the nuances of particular GP levels. This specificity suggests a degree of customization in the feature sets, where each GP level may present unique characteristics that require a tailored approach to feature selection. These findings



suggest that for remote sensing tasks like vegetation classification, a focused set of well-chosen features can significantly enhance model performance by capturing essential information without overburdening the model with redundant data and that this feature set may change depending on the spatial resolution of the available imagery.

The comparative performance analysis between SVM and RF models on various GP levels highlights a significant observation: GP2 outperforms other levels regarding model accuracy and balanced F1-scores. With an accuracy of 0.68, GP2 features align exceptionally well with the classification objective of discerning the relative abundance of spotted knapweed. Intriguingly, the feature set optimized for GP2 is the most compact among all GP levels, consisting of only four features. These are:

- Mean NIR-band reflectance, which is critical for assessing vegetation health as it is strongly absorbed by healthy vegetation.
- Mean red-edge band reflectance, which is a sensitive indicator of chlorophyll content and plant stress.
- Mean green-band entropy, which is a metric of the variability/homogeneity of the green-band reflectance, which may relate to the structure of the vegetative canopy or the presence of different plant species in close proximity to each other.
- Mean NDVI entropy, which is a metric of variability of the NDVI (itself a measure of vegetative vigour) within a metapixel and may relate to sparse canopies through which bare soil can be seen.

The four features that comprise the optimal GP2 feature set leverage data from all four of the spectral bands measured by the Sequoia multi-spectral imager (red band data is included in the NDVI). The mean reflectance values in the Red-edge, and NIR bands are particularly effective because these bands are highly responsive to the presence of vegetation. The GLCM-based entropy features (ENT-green and ENT-NDVI) are likely to capture the unique textural patterns associated with spotted knapweed's sparse canopy and/or groups of heterogeneous plants in close proximity. The irregularity in canopy cover

can create shadows and allow glimpses of the sub-canopy vegetation or bare soil, contributing to a diverse texture signature.

The four features that comprise the GP2 optimal feature set construct a classifier that outperforms classifiers built using data aggregated at different levels of the GPs or multi-scale data from all of the GP levels combined. This suggests that the models built with GP2 data are leveraging both the spectral signatures of the vegetation and the textural context provided by the surrounding environment, which includes shadows and sub-canopy elements. The optimal feature set at GP2 appears to strike a balance, improving classification by reducing the signal-to-noise ratio through spatial averaging in the spectral data. This improvement to the signal-to-noise ratio would not only emphasize the spectral signatures but also enhance the classifier's ability to differentiate between spotted knapweed and other vegetation. It also opens up the possibility of refining the feature selection process to hone in on those attributes that most effectively capture the characteristics of the target species, potentially leading to even more streamlined and efficient models.

The GP2 features correspond to a GSD of 11.6 cm, which is 4 times larger than the original image resolution of 2.9 cm. This specific GSD appears to strike a balance between detail and abstraction, providing an optimal scale for the classification task at hand. Data at finer resolutions (lower GSD) may contain more noise due to measurement error or overly intricate spatial details that could complicate the classification process, whereas coarser resolutions (higher GSD) may lack the requisite spatial detail for distinguishing between different classes. The results suggest that, within the context of this study, smoothing the data to 11.6 cm spatial resolution increases the signal-to-noise ratio of the input features, allowing the classifiers to produce more accurate results with the fewest number of descriptive features compared to the other feature sets considered in this work.

The results obtained from concatenating GP levels, however, have led to an unexpected outcome that contrasts with the findings of Roberti de Siqueira et al. (2013), who showed that a multi-scale representation of image features often produces higher-

performing classifiers. The anticipated advantage of a multi-scale feature set that leverages information from various scales did not materialize. Instead, the GPs models exhibited diminished accuracy, with the RF classifier achieving only 0.38 and the SVM classifier faring slightly better at 0.41. Such outcomes suggest that the multi-scale concatenation diluted rather than enriched the discriminative capability inherent to the features at individual levels. This dilution effect implies that the distinct and nuanced patterns captured by each GP level's features, which might be critical for effective classification, lose their impact when combined. This reduction in performance highlights a potential discrepancy between theoretical expectations and empirical realities, suggesting that the synergistic potential of concatenated multi-scale features may not always hold true across different datasets or classification frameworks. It is also possible that recursive feature selection is not able to identify a truly optimal feature set given the volume and multi-collinearity of these features. Recursive feature selection is a greedy feature optimization heuristic that could lead to degraded performance if many features have similar importance metrics. This could be overcome using a global optimization method like a genetic algorithm (Goldberg, 1994).

In conclusion, this study underscores the significance of selecting an appropriate scale, especially when integrating data sources like GLCM and GP. The superior performance at the 11.6 cm GSD demonstrates the critical role of resolution in image classification tasks.

### Mapping Spotted Knapweed at Site 3

To see the impact of selecting the optimal scale-space representation of the RPAS-acquired imagery for image classification, maps of the relative abundance of spotted knapweed are from spectral data at GP0 and GP2 levels of aggregation collected at Site 3 on July 4. These datasets were withheld from the model-building process, and labeled examples were only utilized to calculate the classifiers' performance metrics (e.g. in Table 12).

An aerial photograph of Site 3 taken during the July 4 imaging flight is shown in Figure 30. This figure shows that there is a diagonal pattern of green vegetative growth that

slopes from the top left to the bottom right. Dense green vegetation also dominates the top right and bottom left corners of the image., while a strip of reddish-grey vegetation runs through the middle of the image. The dark green vegetation is mostly spotted knapweed interspersed with other vegetation, while the reddish-grey vegetation is a patch of cheatgrass (*Bromus tectorum* L), another invasive species common in British Columbia's grasslands.

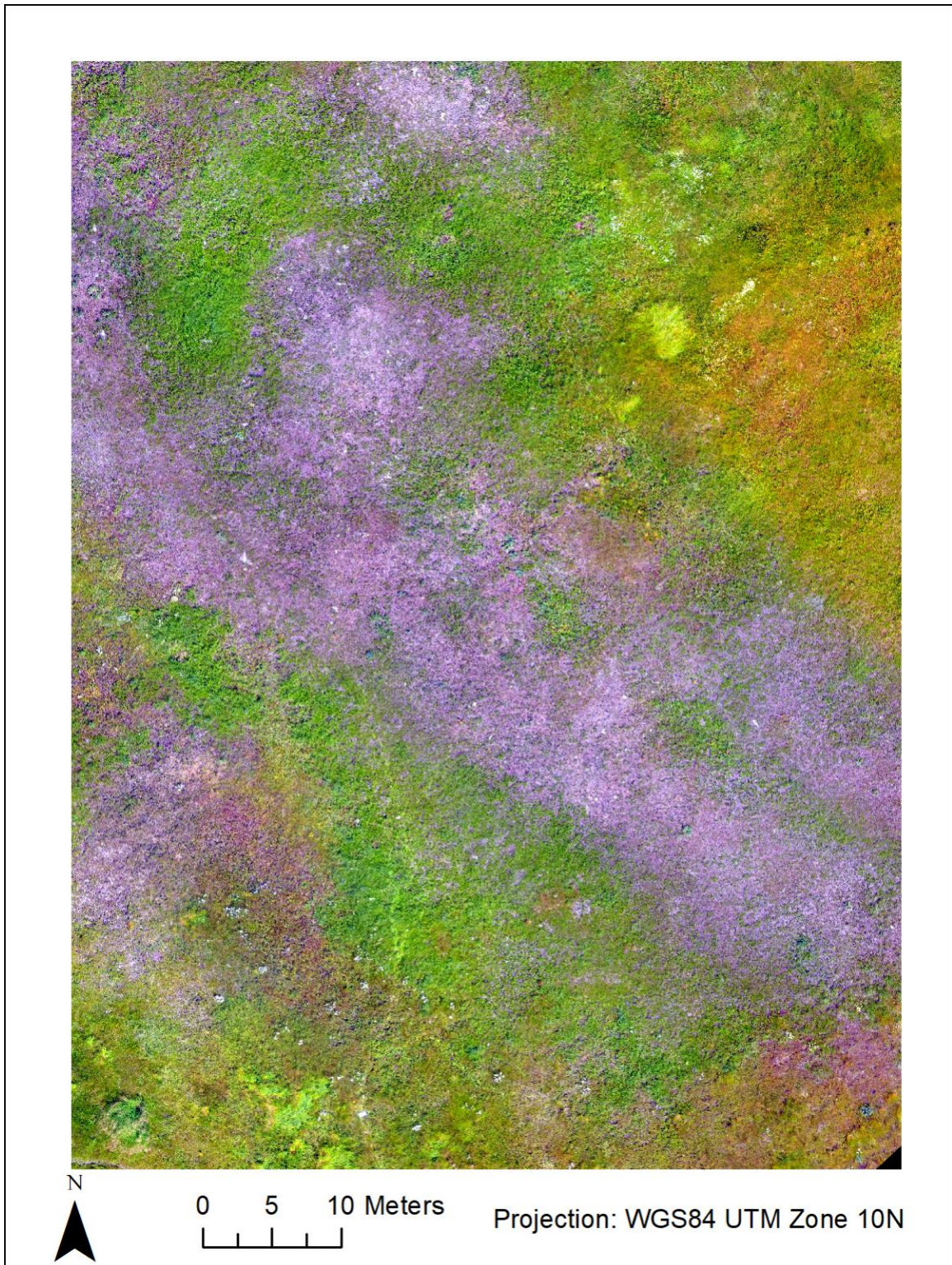
Figure 31 was constructed by applying the GP0 RF classifier to the Site 3 data. This classifier replicates the results presented by Baron and Hill (2020). This figure has a high degree of speckling (i.e. single metapixels with a category that is different from the surrounding metapixels) in the none and moderate categories. The VNIR image in Figure 30 does not reveal much patchiness, so this speckle is likely an artifact of the classifier. The precision of the GP0 for classifying Moderate and None categories of spotted knapweed is 0 and 0.46, so this speckling is likely due to false positive classifications. Furthermore, the recall of the moderate category is 0, which gives no confidence in these classifications.

Figure 32 was constructed by applying the GP2 RF classifier to the Site 3 data. Based on the results of this study, this classifier uses the optimal scale-space representation of the image data. This image exhibits much less speckle than Figure 31. Instead, there is more contiguity between pixels in each class forming larger aggregate groupings which follow the diagonal patterning visible in VNIR image (Figure 30). The precision of the moderate and none classes for this classifier is much higher than for the GP0 RF classifier, which leads to more confidence in these classifications, though the recall of the moderate class is still quite low. The map of spotted knapweed made with the GP2 RF classifier (Figure 32) shows smoother transitions between different abundance classes, whereas the transitions between classes in the map made by the GP0 RF classifier are abrupt. This result suggests that the GP2 scale-space representation of the image data better describes the transitions in knapweed distribution.

When comparing the predicted area of spotted knapweed abundance, the GP0 RF model (Figure 31) classifies 8,358 square meters as High, 2,926 square meters as

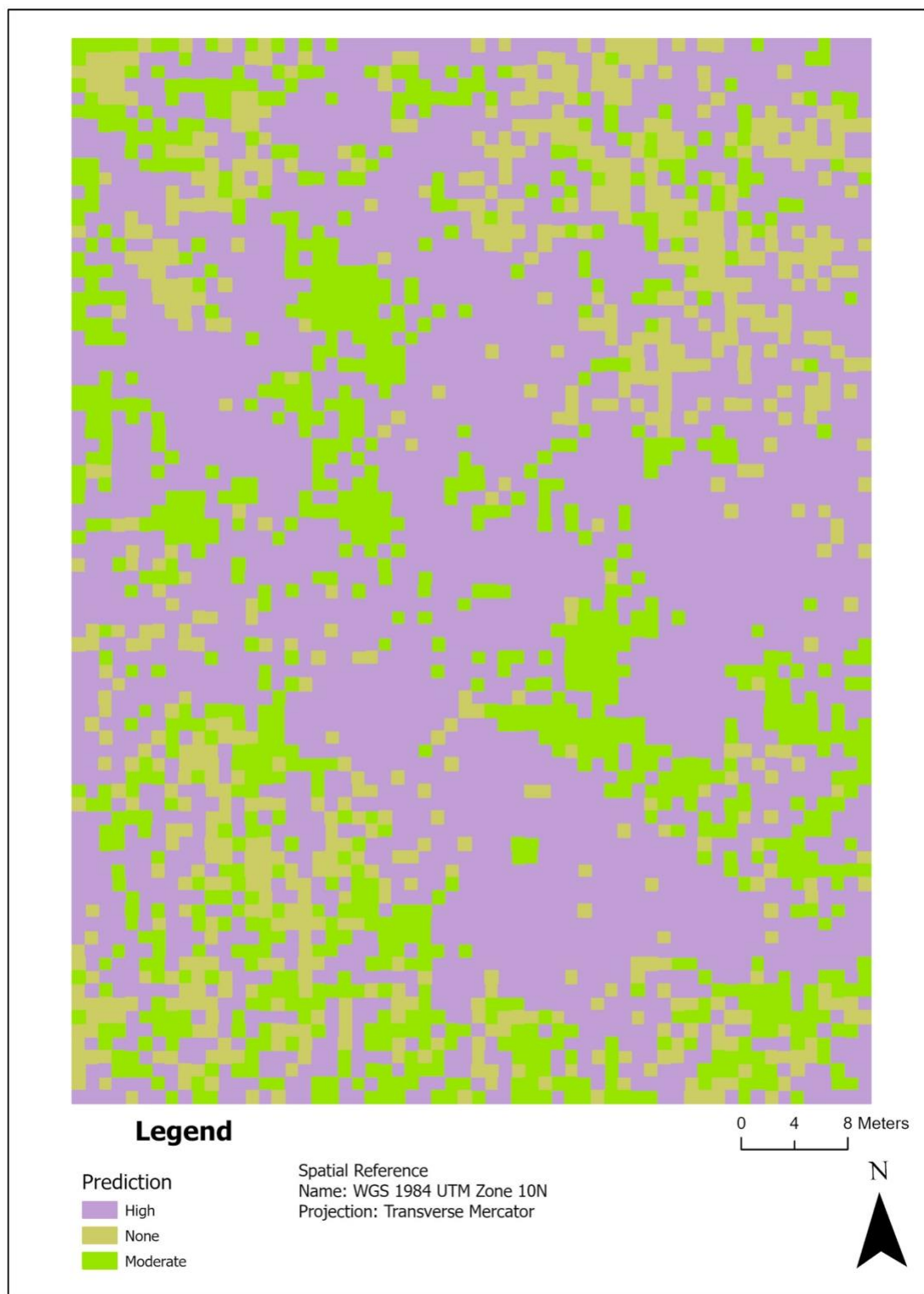
Moderate, and 2,636 square meters as None, whereas the GP2 RF model (Figure 32) classifies 20,288 square meters as High, 25,984 square meters as Moderate, and 9,408 square meters as None. The adoption of the GP2 data suggests that when the image data is represented at a scale that incorporates a moderate level of smoothing—enough to reduce noise and enhance the meaningful signal—it results in a more accurate depiction of the knapweed's spatial distribution. This scale-space optimization allows the GP2 RF model to capture essential characteristics of the knapweed's presence, which might be missed at the finer, less-smoothed GP0 level. The more pronounced smoothing inherent in the GP2 data likely helps to suppress irrelevant variations, thereby strengthening the model's ability to detect the true signal of knapweed abundance. This optimized scale-space representation is key to producing a more reliable and comprehensive map of knapweed distribution, which is crucial for effective monitoring and management of this invasive species.





*Figure 30: True-colour image from flight data collected at field site 3 on July 4, 2018.*





*Figure 31: RF Classification map generated using GLCM-GPO meta pixel-based image analysis, illustrating the relative abundance of spotted knapweed.*



*Figure 32: RF Classification map generated using GLCM-GP2 meta pixel-based image analysis, illustrating the relative abundance of spotted knapweed. This level demonstrates the highest accuracy.*



## Conclusion

This study explored scale-space representations of remote-sensed images to identify optimal features and their relationship with spatial scale, contributing to the understanding of how these representations affect the accuracy of vegetation prediction models.

The identification of mean NIR reflectance and mean red-edge reflectance as significant features across all GP levels is a pivotal finding, emphasizing their importance in capturing the distinctive spectral characteristics of vegetation. These features are particularly sensitive to chlorophyll content, a key indicator of plant health, and the cell structure of vegetation, which is crucial for differentiating between various plant species and conditions. That these two features were selected across different scale-space representations of the image data indicates their importance for identifying and quantifying the relative abundance of spotted knapweed.

Upon examining the classification models, including both RF and SVM, it is evident that the GP2 level exhibits superior performance in terms of accuracy. Notably, the Ground Sample Distance (GSD) of the imagery from the GP2 level, at 11.6 cm, contrasts with the 2.9 cm GSD of the raw imagery. This difference suggests that the smoothing inherent in the GP2 data, which reduces noise and minor variations that may not be relevant to the classification task, likely contributes to improved performance. The GP2 level, with its moderate smoothing, seems to strike an optimal balance between preserving critical information and reducing extraneous detail that could confuse the model. These results carry significant implications for remote sensing analysis. They challenge the conventional approach that typically relies on the raw sensor resolution to dictate the scale of analysis, suggesting instead the potential benefits of scale-space analysis. By optimizing the scale of feature extraction, it is possible to enhance the accuracy of vegetation classification models. Moreover, our findings refute the common assumption that lower spatial-resolution data inherently yields inferior results, which drives RPAS-based remote sensing surveys to be conducted to achieve the highest spatial resolution data that is practicable to acquire. In fact, the enhanced performance at the

GP2 level implies that a compromise in spatial resolution does not necessarily equate to a loss in the quality of analytical outcomes. Consequently, the RPAS data acquisition could have been conducted at a higher elevation above the land surface—around 200 m (rather than 30 m)—resulting in an imagery resolution of 11.6 cm. This adjustment in flight height would have increased the overall size of the scene capture in each RPAS-acquired image, which would have reduced the total number of images that needed to be captured to complete the survey, thereby increasing the speed with which the data was acquired. Such an approach would enable the imaging of larger areas within the same flight time, significantly improving the operational efficiency of the vegetation monitoring efforts.

# Chapter 4 Conclusion & Future works

## Conclusion

The principal aim of this research was to explore the value of scale-space representations of image features to predict the abundance of spotted knapweed within grassland ecosystems. Specifically, this work sought to determine if there was an optimal spatial resolution for image features or if multi-scale representations of features could improve the prediction of spotted knapweed abundance using spectral and grey-scale colocation matrix (GLCM)-based textural features. The Gaussian Pyramid (GP) method was instrumental in determining the spatial resolution that encapsulates the critical data for robust predictive modeling. Notably, this work shows that image features derived from the second level of the gaussian pyramid (GP2 level) produced classifiers that outperformed classifiers trained using the base-resolution image data or image data aggregated to other spatial scales. At the GP2 level, features are derived from image data with 4-times lower spatial resolution than the original data. Interestingly, this research showed that classifiers trained using GP2 data produced better results than classifiers trained using multi-resolution data from all levels in the GP. This latter result contradicts the work of Roberti de Siqueira et al. (2013), who showed that a multi-scale representation of image features often produces higher-performing classifiers. I suspect that this result may be due to the relatively few training examples ( $N=150$ ) compared to the number of features across the entire scale space ( $N=420$ ).

Contrary to the typical remote-sensing approach that equates higher spatial resolution with improved analytical performance, our study proposes that an intermediate resolution, as represented by GP2, may provide a more accurate reflection of ground realities by filtering out noise and minor variations that do not contribute to the species' identification.

The implications for grassland management are profound. Accurately predicting the abundance of spotted knapweed, a virulent invasive species, enables resource managers to deploy control treatments more judiciously, concentrating on areas heavily invaded and conserving efforts in regions with lower incidence. This study illustrates scale-space analysis's potential to increase invasive species mapping accuracy and bolster the

efficacy of ecological management tactics. More accurate invasive species mapping significantly contributes to grassland management in the following areas:

- **Early Detection and Rapid Response:** The model's precision in identifying spotted knapweed is pivotal for detecting invasions early, crucial for rapid response actions to halt the spread before establishment.
- **Precision Management:** Precise spatial estimates of spotted knapweed abundance allow for targeted management. This enables more efficient allocation of resources like herbicides, labor for manual removal, and biocontrol agents, focusing on priority areas and reducing non-target impacts.
- **Monitoring Treatment Efficacy:** Post-treatment monitoring is vital for evaluating management success. Accurate maps of spotted knapweed abundance derived from pre- and post-intervention remote sensing imagery can be compared to compute intervention success metrics.
- **Adaptive Management:** Invasive species management is dynamic, requiring adaptation to change. The invasive species mapping method developed in this work supports continuous monitoring, yielding data to inform adaptive management decisions and strategy adjustments.
- **Habitat Restoration Planning:** By delineating spotted knapweed's spatial distribution, the model supports restoration planning, identifying reintroduction sites for native species and guiding restoration efforts to restore native plant community structure and ecosystem functions.
- **Cost-Effective Surveying:** The case study explored in Chapter 3 revealed that the optimal image resolution for the mapping of spotted knapweed abundance was four-times larger than the resolution at which the imagery was acquired. This result suggests that the images could have been acquired at a much higher flight level. Flying a remote sensing platform (e.g., a remotely piloted aircraft system, RPAS) at a higher flight level will enable the acquisition of images with a larger field of view, reducing the time required to image a fixed spatial extent. Thus, by tuning the flight level to the optimal spatial resolution for image analysis, larger areas can be surveyed more

economically, enabling more frequent surveys and better-informed management decisions.

In conclusion, the application of this model within grassland management programs can significantly enhance the effectiveness of spotted knapweed control and mitigation efforts. Its integration into routine monitoring practices represents a proactive step towards sustainable management of grassland ecosystems.

While the model devised in this study is tailored to a specific species and may not directly translate to other species, the underlying modeling framework I have established has the potential for broader application. To adapt this approach for a different species, it would be essential to fine-tune the features and hyperparameters to align with the unique spectral characteristics and spatial distribution patterns of the new species. Additionally, this research indicates that determining the optimal scale for analysis by identifying the most effective GP level is crucial. With these customizations, the updated model could then be utilized to map the presence of the new target species across extended regions.

Future work should focus on refining the categories used to define the relative abundance of spotted knapweed through additional data collection and stratified spatial sampling methods, evaluating the generalizability of the models developed using this method over larger spatial extents, and extending its application to other invasive species and habitat types. Additionally, the development of user-friendly tools and interfaces for land managers to utilize this model can facilitate its widespread adoption in conservation and land management sectors.

## Future Work - Expanding Methodologies

As we progress, multiple prospects for continued research and enhancement in this field emerge:

### Generative Adversarial Networks for Training Data Augmentation

The exploration of generative adversarial networks (GANs) to supplement the training dataset constitutes a significant direction for future research. The necessity for a balanced training set in this study was imperative to avoid category bias. To achieve a balanced

training set, however, the data available for training we down sampled selecting only as many examples of each category as were available in the minority class. This resulted in a small training set size of 150 examples. With such a small number of training examples, the machine learning classifiers used in this study (i.e. random forest and support vector machine) would not be able to utilize the information available from a large set of input features, such as the 84 features used in the single GP-level models.

A GAN is a framework for generative artificial intelligence, in which two artificial neural networks (ANNs) compete against each other. One ANN, the generator, seeks to create synthetic examples that match a pattern in input data, while the other Ann, the discriminator, seeks to discriminate between examples synthesized by the generator and those existing in the input set. The introduction of GANs could address the limitation of a small training set size by generating synthetic yet plausible training examples that mimic the patterns in the true training examples. Employing a GAN to oversample the true training data could enhance the training process without the associated risk of overfitting that sample duplication entails. This technique allows for the expansion of the training set size, potentially increasing it to 300 samples, evenly distributed across classifications. Such expansion could bolster the model's generalization and fortify the predictions of spotted knapweed prevalence.

### Feature Optimization through Data Compression

Data compression methods like Principal Component Analysis (PCA) and autoencoders represent an avenue for refining feature selection in future research endeavors. Although recursive feature elimination with cross-validation (RFECV) has demonstrated efficacy in identifying an optimal feature set in this research, PCA and autoencoders present alternative means for isolating pivotal features within a dataset.

PCA can decrease the dimensionality of data while retaining significant variance, which may reveal the data's intrinsic structure. Autoencoders, which are specialized ANNs, use a deep network to encode input data efficiently in a central encoding layer and subsequently reconstruct output from this encoded information. Training autoencoders

can lead to the extraction of a feature set that encapsulates the essential details necessary for precise predictions.

These techniques are especially beneficial in high-dimensional data scenarios, aiming to distill the feature set without forfeiting pertinent information critical to the classification task. Compressing the feature space could aid in distilling the relevant multi-scale features present throughout the scale-space resulting in a more streamlined set of input features for evaluation with RFECV. Not only will this reduce the computational demand of classifier training, but it could also result in the learning of more accurate classifiers. Such advancements could markedly improve the mapping and monitoring processes for invasive species like spotted knapweed in grassland ecosystems.

The assimilation of these sophisticated methodologies is anticipated to expand the toolkit available for remote sensing analyses and species distribution modeling. Future studies can build upon the groundwork established by this thesis to further enhance our proficiency in ecological conservation and management, safeguarding the fragile equilibriums in our natural habitats.

### Other suggestions

Deep learning approaches, such as convolutional neural networks (CNNs), and their performance analysis across various spatial resolutions could provide additional insights and might even lead to the automation of identifying the optimal spatial scale for feature extraction.

Exploring scale-space representations of other image datasets, like hyperspectral imagery, could enrich the dataset and potentially improve classification models. With continuous advancements in sensor technology, assessing their performance across different scales will be essential, including the impact of new sensors on scale-space representation and feature selection. Future research should also consider the operational implications, such as the effects of altering flight altitude on the efficiency and cost-effectiveness of RPAS-based imaging surveys.

The concepts derived from this study might also have more extensive applications in spatial analysis, extending to urban planning, disaster management, and geological



surveys. By delving into these areas of future work, the field can evolve toward more sophisticated, precise, and efficient environmental monitoring and analysis methods. The ultimate aim is to establish a set of best practices for scale-space representation in remote sensing that can be tailored to diverse situations and applications.

# REFERENCES

- Adelson, E.H., Anderson, C.H., Bergen, J.R., Burt, P.J., Ogden, J.M., 1984. Pyramid methods in image processing. *RCA engineer* 29, 33–41.
- Adi, S., Pristyanto, Y., Sunyoto, A., 2019. The best features selection method and relevance variable for web phishing classification. Presented at the 2019 International Conference on Information and Communications Technology (ICOIACT), IEEE, pp. 578–583.
- Andrew, M.E., Ustin, S.L., 2009. Habitat suitability modelling of an invasive plant with advanced remote sensing data. *Diversity and Distributions* 15, 627–640. <https://doi.org/10.1111/j.1472-4642.2009.00568.x>
- Ataei, M., Osanloo, M., 2004. Using a combination of genetic algorithm and the grid search method to determine optimum cutoff grades of multiple metal deposits. *International Journal of Surface Mining, Reclamation and Environment* 18, 60–78.
- Baraldi, A., Pannigiani, F., 1995. An investigation of the textural characteristics associated with gray level cooccurrence matrix statistical parameters. *IEEE Transactions on Geoscience and Remote Sensing* 33, 293–304. <https://doi.org/10.1109/TGRS.1995.8746010>
- Baron, J., Hill, D.J., 2020. Monitoring grassland invasion by spotted knapweed (*Centaurea maculosa*) with RPAS-acquired multispectral imagery. *Remote Sensing of Environment* 249, 112008. <https://doi.org/10.1016/j.rse.2020.112008>
- Baron, J., Hill, D.J., Elmiligi, H., 2018. Combining image processing and machine learning to identify invasive plants in high-resolution images. *International Journal of Remote Sensing* 39, 5099–5118. <https://doi.org/10.1080/01431161.2017.1420940>
- Baron, J.P.J., 2020. Mapping invasive plants using RPAS and remote sensing.
- Belgiu, M., Drăguț, L., 2016. Random forest in remote sensing: A review of applications and future directions. *ISPRS journal of photogrammetry and remote sensing* 114, 24–31.
- Bradley, B.A., 2014. Remote detection of invasive plants: a review of spectral, textural and phenological approaches. *Biol Invasions* 16, 1411–1425. <https://doi.org/10.1007/s10530-013-0578-9>
- Bradski, G., 2000. The openCV library. *Dr. Dobb's Journal: Software Tools for the Professional Programmer* 25, 120–123.
- Breiman, L., 2001. Random Forests. *Machine Learning* 45, 5–32. <https://doi.org/10.1023/A:1010933404324>
- Carlson, T.N., Ripley, D.A., 1997. On the relation between NDVI, fractional vegetation cover, and leaf area index. *Remote Sensing of Environment* 62, 241–252. [https://doi.org/10.1016/S0034-4257\(97\)00104-1](https://doi.org/10.1016/S0034-4257(97)00104-1)
- Chang, W., Liu, Y., Xiao, Y., Yuan, X., Xu, X., Zhang, S., Zhou, S., 2019. A machine-learning-based prediction method for hypertension outcomes based on medical data. *Diagnostics* 9, 178.
- Chaudhuri, P., Marron, J.S., 2000. Scale space view of curve estimation. *The Annals of Statistics* 28, 408–428. <https://doi.org/10.1214/aos/1016218224>

- Cherkassky, V., Ma, Y., 2004. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks* 17, 113–126. [https://doi.org/10.1016/S0893-6080\(03\)00169-2](https://doi.org/10.1016/S0893-6080(03)00169-2)
- Chomat, O., Colin de Verdière, V., Crowley, J.L., 2001. Recognizing goldfish? or Local scale selection for recognition techniques. *Robotics and Autonomous Systems, Seventh Symposium on Intelligent Robotic Systems - SIRS'99* 35, 191–200. [https://doi.org/10.1016/S0921-8890\(01\)00124-5](https://doi.org/10.1016/S0921-8890(01)00124-5)
- Coelho, L.P., 2013. Mahotas: Open source software for scriptable computer vision 1, e3. <https://doi.org/10.5334/jors.ac>
- Connors, R.W., Harlow, C.A., 1980. A Theoretical Comparison of Texture Algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-2*, 204–222. <https://doi.org/10.1109/TPAMI.1980.4767008>
- Cortes, C., Vapnik, V., 1995. Support-vector networks. *Machine learning* 20, 273–297.
- Deriche, R., 1993. Recursively implementing the Gaussian and its derivatives (PhD Thesis). INRIA.
- Diagne, C., Leroy, B., Vaissière, A.-C., Gozlan, R.E., Roiz, D., Jarić, I., Salles, J.-M., Bradshaw, C.J.A., Courchamp, F., 2021. High and rising economic costs of biological invasions worldwide. *Nature* 592, 571–576. <https://doi.org/10.1038/s41586-021-03405-6>
- Dorigo, W., Lucieer, A., Podobnikar, T., Čarni, A., 2012. Mapping invasive *Fallopia japonica* by combined spectral, spatial, and temporal analysis of digital orthophotos. *International Journal of Applied Earth Observation and Geoinformation* 19, 185–195. <https://doi.org/10.1016/j.jag.2012.05.004>
- Duncan, P., Podest, E., Esler, K.J., Geerts, S., Lyons, C., 2023. Mapping Invasive Herbaceous Plant Species with Sentinel-2 Satellite Imagery: *Echium plantagineum* in a Mediterranean Shrubland as a Case Study. *Geomatics* 3, 328–344. <https://doi.org/10.3390/geomatics3020018>
- Dvořák, P., Müllerová, J., Bartaloš, T., Brůna, J., 2015. Unmanned aerial vehicles for alien plant species detection and monitoring. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 40, 83–90.
- Efron, B., Tibshirani, R.J., 1994. *An introduction to the bootstrap*. CRC press.
- Elith, J., 2019. 15-Machine learning, random forests, and boosted regression trees. *Quantitative analyses in wildlife science* 281.
- Erickson, B.J., Korfiatis, P., Akkus, Z., Kline, T.L., 2017. Machine learning for medical imaging. *Radiographics* 37, 505–515.
- Florack, L.M.J., 1997. *Image Structure (Computational Imaging and Vision #10)*. KluwerAcademicPublishers, Dordrecht,TheNetherlands.
- Florack, L.M.J., ter Haar Romeny, B.M., Koenderink, J.J., Viergever, M.A., 1992. Scale and the differential structure of images. *Image and Vision Computing* 10, 376–388. [https://doi.org/10.1016/0262-8856\(92\)90024-W](https://doi.org/10.1016/0262-8856(92)90024-W)
- Foody, G.M., Mathur, A., 2004. A relative evaluation of multiclass image classification by support vector machines. *IEEE Transactions on Geoscience and Remote Sensing* 42, 1335–1343. <https://doi.org/10.1109/TGRS.2004.827257>
- Foster, J.G., Ploughe, L.W., Akin-Fajiye, M., Singh, J.P., Bottos, E., Van Hamme, J., Fraser, L.H., 2020. Exploring trophic effects of spotted knapweed (*Centaurea stoebe* L.) on

- arthropod diversity using DNA metabarcoding. *Food Webs* 24, e00157. <https://doi.org/10.1016/j.fooweb.2020.e00157>
- Fushiki, T., 2011. Estimation of prediction error by using K-fold cross-validation. *Stat Comput* 21, 137–146. <https://doi.org/10.1007/s11222-009-9153-8>
- Gaskin, J.F., Espeland, E., Johnson, C.D., Larson, D.L., Mangold, J.M., McGee, R.A., Milner, C., Paudel, S., Pearson, D.E., Perkins, L.B., Prosser, C.W., Runyon, J.B., Sing, S.E., Sylvain, Z.A., Symstad, A.J., Tekiela, D.R., 2021. Managing invasive plants on Great Plains grasslands: A discussion of current challenges. *Rangeland Ecology & Management*, Great Plains 78, 235–249. <https://doi.org/10.1016/j.rama.2020.04.003>
- Gholizadeh, H., Friedman, M.S., McMillan, N.A., Hammond, W.M., Hassani, K., Sams, A.V., Charles, M.D., Garrett, D.R., Joshi, O., Hamilton, R.G., Fuhlendorf, S.D., Trowbridge, A.M., Adams, H.D., 2022. Mapping invasive alien species in grassland ecosystems using airborne imaging spectroscopy and remotely observable vegetation functional traits. *Remote Sensing of Environment* 271, 112887. <https://doi.org/10.1016/j.rse.2022.112887>
- Goldberg, D.E., 1994. Genetic and evolutionary algorithms come of age. *Communications of the ACM* 37, 113–120.
- Gonzalez, R.C., Woods, R.E., Hall, P.P., 2008. Digital Image Processing Third Edition Pearson International Edition Prepared by Pearson Education. *Journal of Biomedical Optics* 14, 029901.
- Guan, H., Li, J., Chapman, M., Deng, F., Ji, Z., Yang, X., 2013. Integration of orthoimagery and lidar data for object-based urban thematic mapping using random forests. *International Journal of Remote Sensing* 34, 5166–5186.
- Haddad, R.A., Akansu, A.N., 1991. A class of fast Gaussian binomial filters for speech and image processing. *IEEE Transactions on Signal Processing* 39, 723–727.
- Hall-Beyer, M., 2017. Practical guidelines for choosing GLCM textures to use in landscape classification tasks over a range of moderate spatial scales. *International Journal of Remote Sensing* 38, 1312–1338.
- Haralick, R.M., Shanmugam, K., 1973. Computer Classification of Reservoir Sandstones. *IEEE Transactions on Geoscience Electronics* 11, 171–177. <https://doi.org/10.1109/TGE.1973.294312>
- Haralick, R.M., Shanmugam, K., Dinstein, I., 1973. Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics SMC-3*, 610–621. <https://doi.org/10.1109/TSMC.1973.4309314>
- Harris, C.R., Millman, K.J., Walt, S.J. van der, Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N.J., Kern, R., Picus, M., Hoyer, S., Kerkwijk, M.H. van, Brett, M., Haldane, A., Río, J.F. del, Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., Oliphant, T.E., 2020. Array programming with NumPy. *Nature* 585, 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
- Hastie, T., Tibshirani, R., Friedman, J.H., Friedman, J.H., 2009. The elements of statistical learning: data mining, inference, and prediction. Springer.

- Hawryło, P., Bednarz, B., Wężyk, P., Szostak, M., 2018. Estimating defoliation of Scots pine stands using machine learning methods and vegetation indices of Sentinel-2. *European Journal of Remote Sensing* 51, 194–204.
- Henkel, R.D., 1995. Segmentation in scale space, in: Hlaváč, V., Šára, R. (Eds.), *Computer Analysis of Images and Patterns, Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp. 41–48. [https://doi.org/10.1007/3-540-60268-2\\_278](https://doi.org/10.1007/3-540-60268-2_278)
- Hill, D., Pypker, T., Church, J., 2020. Applications of Unpiloted Aerial Vehicles (UAVs) in Forest Hydrology. *Remote Sensing*.
- Huang, C., Geiger, E.L., 2008. Climate anomalies provide opportunities for large-scale mapping of non-native plant abundance in desert grasslands. *Diversity and Distributions* 14, 875–884. <https://doi.org/10.1111/j.1472-4642.2008.00500.x>
- Hummel, R.A., Kimia, B., Zucker, S.W., 1987. Deblurring Gaussian blur. *Computer Vision, Graphics, and Image Processing* 38, 66–80. [https://doi.org/10.1016/S0734-189X\(87\)80153-6](https://doi.org/10.1016/S0734-189X(87)80153-6)
- Ikonomakis, M., Kotsiantis, S., Tampakas, V., 2005. Text classification using machine learning techniques. *WSEAS transactions on computers* 4, 966–974.
- Ishii, J., Washitani, I., 2013. Early detection of the invasive alien plant *Solidago altissima* in moist tall grassland using hyperspectral imagery. *International Journal of Remote Sensing* 34, 5926–5936. <https://doi.org/10.1080/01431161.2013.799790>
- Kalitzin, S.N., ter Haar Romeny, B., Viergever, M., 1997. On topological deep-structure segmentation, in: *Proceedings of International Conference on Image Processing*. Presented at the Proceedings of International Conference on Image Processing, pp. 863–866 vol.2. <https://doi.org/10.1109/ICIP.1997.638633>
- Kamalov, F., Gurrib, I., Rajab, K., 2021. Financial forecasting with machine learning: price vs return. Kamalov, F., Gurrib, I. & Rajab, K.(2021). *Financial Forecasting with Machine Learning: Price Vs Return*. *Journal of Computer Science* 17, 251–264.
- Kekre, H.B., Thepade, S.D., Sarode, T.K., Suryawanshi, V., 2010. Image Retrieval using Texture Features extracted from GLCM, LBG and KPE. *International Journal of Computer Theory and Engineering* 2, 695.
- Kelkar, K.M., Bakal, J.W., 2020. Hyper Parameter Tuning of Random Forest Algorithm for Affective Learning System, in: *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*. Presented at the 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), pp. 1192–1195. <https://doi.org/10.1109/ICSSIT48917.2020.9214213>
- Khare, S., Latifi, H., Ghosh, S.K., 2018. Multi-scale assessment of invasive plant species diversity using Pléiades 1A, RapidEye and Landsat-8 data. *Geocarto International* 33, 681–698. <https://doi.org/10.1080/10106049.2017.1289562>
- Klusowski, J.M., 2018. Complete analysis of a random forest model. arXiv preprint arXiv:1805.02587.
- Koenderink, J.J., 1984. The structure of images. *Biological Cybernetics* 50, 363–370. <https://doi.org/10.1007/BF00336961>
- Konlambigue, S., Pothin, J.-B., Honeine, P., Bensrhair, A., 2018. Fast and Accurate Gaussian Pyramid Construction by Extended Box Filtering, in: *2018 26th European Signal Processing Conference (EUSIPCO)*. Presented at the 2018 26th European Signal

- Processing Conference (EUSIPCO), pp. 400–404.  
<https://doi.org/10.23919/EUSIPCO.2018.8553321>
- Kuhn, M., Johnson, K., 2013. Applied predictive modeling. Springer.
- Kuijper, A., Florack, L.M.J., Viergever, M.A., 2003. Scale Space Hierarchy. *Journal of Mathematical Imaging and Vision* 18, 169–189.  
<https://doi.org/10.1023/A:1022168617945>
- Lake, T.A., Briscoe Runquist, R.D., Moeller, D.A., 2022. Deep learning detects invasive plant species across complex landscapes using Worldview-2 and PlanetScope satellite imagery. *Remote Sensing in Ecology and Conservation* 8, 875–889.  
<https://doi.org/10.1002/rse2.288>
- Lehmann, J.R., Prinz, T., Ziller, S.R., Thiele, J., Heringer, G., Meira-Neto, J.A., Buttschardt, T.K., 2017. Open-source processing and analysis of aerial imagery acquired with a low-cost unmanned aerial system to support invasive plant management. *Frontiers in Environmental Science* 5, 44.
- Li, J., Li, D., Zhang, G., Xu, H., Zeng, R., Luo, W., Yu, Y., 2019. Study on extraction of foreign invasive species *Mikania micrantha* based on unmanned aerial vehicle (UAV) hyperspectral remote sensing, in: Fifth Symposium on Novel Optoelectronic Detection Technology and Application. Presented at the Fifth Symposium on Novel Optoelectronic Detection Technology and Application, SPIE, pp. 597–605.  
<https://doi.org/10.1117/12.2520027>
- Li, S., Hao, Q., Kang, X., Benediktsson, J.A., 2018. Gaussian Pyramid Based Multiscale Feature Fusion for Hyperspectral Image Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 11, 3312–3324.  
<https://doi.org/10.1109/JSTARS.2018.2856741>
- Lindeberg, T., 2020. Scale Selection, in: *Computer Vision: A Reference Guide*. Springer International Publishing, Cham, pp. 1–14. [https://doi.org/10.1007/978-3-030-03243-2\\_242-1](https://doi.org/10.1007/978-3-030-03243-2_242-1)
- Lindeberg, T., 2013. *Scale-space theory in computer vision*. Springer Science & Business Media.
- Lindeberg, T., 2009. *Scale-Space*. John Wiley & Sons, pp. 2495–2504.
- Lindeberg, T., 1995. Direct estimation of affine image deformations using visual front-end operations with automatic scale selection, in: *Proceedings of IEEE International Conference on Computer Vision*. IEEE, pp. 134–141.
- Lindeberg, T., 1990. Scale-space for discrete signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12, 234–254. <https://doi.org/10.1109/34.49051>
- Malanson, G.P., Walsh, S.J., 2013. A Geographical Approach to Optimization of Response to Invasive Species, in: Walsh, S.J., Mena, C.F. (Eds.), *Science and Conservation in the Galapagos Islands: Frameworks & Perspectives, Social and Ecological Interactions in the Galapagos Islands*. Springer, New York, NY, pp. 199–215.  
[https://doi.org/10.1007/978-1-4614-5794-7\\_12](https://doi.org/10.1007/978-1-4614-5794-7_12)
- Mallmann, C., Zaninni, A., Pereira Filho, W., 2020. Vegetation index based in unmanned aerial vehicle (UAV) to improve the management of invasive plants in Protected Areas, Southern Brazil. Presented at the 2020 IEEE Latin American GRSS & ISPRS Remote Sensing Conference (LAGIRS), IEEE, pp. 66–69.

- Matongera, T.N., Mutanga, O., Dube, T., Sibanda, M., 2017. Detection and mapping the spatial distribution of bracken fern weeds using the Landsat 8 OLI new generation sensor. *International Journal of Applied Earth Observation and Geoinformation* 57, 93–103. <https://doi.org/10.1016/j.jag.2016.12.006>
- McKinney, W., others, 2010. Data structures for statistical computing in python, in: *Proceedings of the 9th Python in Science Conference*. Austin, TX, pp. 51–56.
- Mikolajczyk, K., 2002. Detection of local features invariant to affines transformations (Theses). Institut National Polytechnique de Grenoble - INPG.
- Mingers, J., 1989. An empirical comparison of pruning methods for decision tree induction. *Machine learning* 4, 227–243.
- Misra, P., Yadav, A.S., 2020. Improving the classification accuracy using recursive feature elimination with cross-validation. *Int. J. Emerg. Technol* 11, 659–665.
- Mitchell, J.J., Glenn, N.F., 2009. Subpixel abundance estimates in mixture-tuned matched filtering classifications of leafy spurge (*Euphorbia esula* L.). *International Journal of Remote Sensing* 30, 6099–6119. <https://doi.org/10.1080/01431160902810620>
- Mohanaiah, P., Sathyanarayana, P., GuruKumar, L., 2013. Image texture feature extraction using GLCM approach. *International journal of scientific and research publications* 3, 1–5.
- Mountrakis, G., Im, J., Ogole, C., 2011. Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing* 66, 247–259. <https://doi.org/10.1016/j.isprsjprs.2010.11.001>
- Mpinda Ataky, S.T., de Matos, J., Britto, A. de S., Oliveira, L.E.S., Koerich, A.L., 2020. Data Augmentation for Histopathological Images Based on Gaussian-Laplacian Pyramid Blending, in: *2020 International Joint Conference on Neural Networks (IJCNN)*. Presented at the 2020 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. <https://doi.org/10.1109/IJCNN48605.2020.9206855>
- Murphy, K.P., 2012. *Machine learning: a probabilistic perspective*. MIT press.
- Ng, W.-T., Meroni, M., Immitzer, M., Böck, S., Leonardi, U., Rembold, F., Gadain, H., Atzberger, C., 2016. Mapping *Prosopis* spp. with Landsat 8 data in arid environments: Evaluating effectiveness of different methods and temporal imagery selection for Hargeisa, Somaliland. *International Journal of Applied Earth Observation and Geoinformation* 53, 76–89. <https://doi.org/10.1016/j.jag.2016.07.019>
- Nininahazwe, F., Théau, J., Marc Antoine, G., Varin, M., 2023. Mapping invasive alien plant species with very high spatial resolution and multi-date satellite imagery using object-based and machine learning techniques: A comparative study. *GIScience & Remote Sensing* 60, 2190203. <https://doi.org/10.1080/15481603.2023.2190203>
- Olkkonen, H., Pesola, P., 1996. Gaussian Pyramid Wavelet Transform for Multiresolution Analysis of Images. *Graphical Models and Image Processing* 58, 394–398. <https://doi.org/10.1006/gmip.1996.0032>
- O'Mara, F.P., 2012. The role of grasslands in food security and climate change. *Annals of botany* 110, 1263–1270.



- Öztürk, Ş., Akdemir, B., 2018. Application of feature extraction and classification methods for histopathological image using GLCM, LBP, LBGLCM, GLRLM and SFTA. *Procedia computer science* 132, 40–46.
- Pal, M., 2005. Random forest classifier for remote sensing classification. *International Journal of Remote Sensing* 26, 217–222. <https://doi.org/10.1080/01431160412331269698>
- Pearlstine, L., Portier, K.M., Smith, S.E., 2005. Textural Discrimination of an Invasive Plant, *Schinus terebinthifolius*, from Low Altitude Aerial Digital Imagery. *Photogrammetric Engineering & Remote Sensing* 71, 289–298. <https://doi.org/10.14358/PERS.71.3.289>
- Petropoulos, G.P., Arvanitis, K., Sigrimis, N., 2012. Hyperion hyperspectral imagery analysis combined with machine learning classifiers for land use/cover mapping. *Expert systems with Applications* 39, 3800–3809.
- Probst, P., Wright, M.N., Boulesteix, A.-L., 2019. Hyperparameters and tuning strategies for random forest. *WIREs Data Mining and Knowledge Discovery* 9, e1301. <https://doi.org/10.1002/widm.1301>
- Pyšek, P., Richardson, D.M., 2010. Invasive Species, Environmental Change and Management, and Health. *Annual Review of Environment and Resources* 35, 25–55. <https://doi.org/10.1146/annurev-environ-033009-095548>
- Qian, H., Ricklefs, R.E., 2006. The role of exotic species in homogenizing the North American flora. *Ecology Letters* 9, 1293–1298. <https://doi.org/10.1111/j.1461-0248.2006.00982.x>
- Réti, Z., 1995. Deblurring images blurred by the discrete Gaussian. *Applied Mathematics Letters* 8, 29–35. [https://doi.org/10.1016/0893-9659\(95\)00042-O](https://doi.org/10.1016/0893-9659(95)00042-O)
- Roberti de Siqueira, F., Robson Schwartz, W., Pedrini, H., 2013. Multi-scale gray level co-occurrence matrices for texture description. *Neurocomputing, Image Feature Detection and Description* 120, 336–345. <https://doi.org/10.1016/j.neucom.2012.09.042>
- Romeny, B.M.H., 2008. *Front-End Vision and Multi-Scale Image Analysis: Multi-scale Computer Vision Theory and Applications*, written in Mathematica. Springer Science & Business Media.
- Royimani, L., Mutanga, O., Odindi, J., Dube, T., Matongera, T.N., 2019. Advancements in satellite remote sensing for mapping and monitoring of alien invasive plant species (AIPs). *Physics and Chemistry of the Earth, Parts A/B/C, 18th WaterNet/WARFSA/GWPSA Symposium on Integrated Water Resources Development and Management: Innovative Technological Advances for Water Security in Eastern and Southern Africa - Part B* 112, 237–245. <https://doi.org/10.1016/j.pce.2018.12.004>
- Rupasinghe, P.A., Chow-Fraser, P., 2021. Mapping Phragmites cover using WorldView 2/3 and Sentinel 2 images at Lake Erie Wetlands, Canada. *Biol Invasions* 23, 1231–1247. <https://doi.org/10.1007/s10530-020-02432-0>
- Samson, F., Knopf, F., 1994. Prairie Conservation in North America. *BioScience* 44, 418–421. <https://doi.org/10.2307/1312365>

- Sebastian, B., Unnikrishnan, A., Balakrishnan, K., 2012. GREY LEVEL CO-OCCURRENCE MATRICES: GENERALISATION AND SOME NEW FEATURES. *International Journal of Computer Science, Engineering and Information Technology (IJCEIT)* Vol.2, No.2.
- Selvaraj, M.G., Vergara, A., Montenegro, F., Ruiz, H.A., Safari, N., Raymaekers, D., Ocimati, W., Ntamwira, J., Tits, L., Omondi, A.B., 2020. Detection of banana plants and their major diseases through aerial images and machine learning methods: A case study in DR Congo and Republic of Benin. *ISPRS Journal of Photogrammetry and Remote Sensing* 169, 110–124.
- Shafizadeh-Moghadam, H., Asghari, A., Tayyebi, A., Taleai, M., 2017. Coupling machine learning, tree-based and statistical models with cellular automata to simulate urban growth. *Computers, Environment and Urban Systems* 64, 297–308.
- Sheykhmousa, M., Mahdianpari, M., Ghanbari, H., Mohammadimanesh, F., Ghamisi, P., Homayouni, S., 2020. Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13, 6308–6325. <https://doi.org/10.1109/JSTARS.2020.3026724>
- Shiferaw, H., Bewket, W., Eckert, S., 2019. Performances of machine learning algorithms for mapping fractional cover of an invasive plant species in a dryland ecosystem. *Ecology and Evolution* 9, 2562–2574. <https://doi.org/10.1002/ece3.4919>
- Singh, J.P., Kuang, Y., Ploughe, L., Coghill, M., Fraser, L.H., 2022. Spotted knapweed (*Centaurea stoebe*) creates a soil legacy effect by modulating soil elemental composition in a semi-arid grassland ecosystem. *Journal of Environmental Management* 317, 115391.
- Sporring, J., Nielsen, M., Florack, L., Johansen, P., 2013. *Gaussian scale-space theory*. Springer Science & Business Media.
- Strobl, C., Malley, J., Tutz, G., 2009. An introduction to recursive partitioning: Rationale, application, and characteristics of classification and regression trees, bagging, and random forests. *Psychological Methods* 14, 323–348. <https://doi.org/10.1037/a0016973>
- Svetnik, V., Liaw, A., Tong, C., Wang, T., 2004. Application of Breiman's random forest to modeling structure-activity relationships of pharmaceutical molecules. Presented at the Multiple Classifier Systems: 5th International Workshop, MCS 2004, Cagliari, Italy, June 9-11, 2004. *Proceedings* 5, Springer, pp. 334–343.
- Tahir, M.A., Roula, M.A., Bouridane, A., Kurugollu, F., Amira, A., 2003. An FPGA based co-processor for GLCM texture features measurement, in: 10th IEEE International Conference on Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003. Presented at the 10th IEEE International Conference on Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003, pp. 1006-1009 Vol.3. <https://doi.org/10.1109/ICECS.2003.1301679>
- Thessen, A., 2016. Adoption of Machine Learning Techniques in Ecology and Earth Science. *One Ecosystem* 1, e8621. <https://doi.org/10.3897/oneeco.1.e8621>
- Turner, D., Lucieer, A., Watson, C., 2012. An Automated Technique for Generating Georectified Mosaics from Ultra-High Resolution Unmanned Aerial Vehicle (UAV)

- Imagery, Based on Structure from Motion (SfM) Point Clouds. *Remote Sensing* 4, 1392–1410. <https://doi.org/10.3390/rs4051392>
- Underwood, E.C., Ustin, S.L., Ramirez, C.M., 2007. A Comparison of Spatial and Spectral Image Resolution for Mapping Invasive Plants in Coastal California. *Environmental Management* 39, 63–83. <https://doi.org/10.1007/s00267-005-0228-9>
- Ustebay, S., Turgut, Z., Aydin, M.A., 2018. Intrusion Detection System with Recursive Feature Elimination by Using Random Forest and Deep Learning Classifier, in: 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT). Presented at the 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), pp. 71–76. <https://doi.org/10.1109/IBIGDELFT.2018.8625318>
- Valavi, R., Elith, J., Lahoz-Monfort, J.J., Guillera-Arroita, G., 2021. Modelling species presence-only data with random forests. *Ecography* 44, 1731–1742.
- Vapnik, V., 2006. Estimation of dependences based on empirical data. Springer Science & Business Media.
- Wibawa, M.S., Novianti, K.D.P., 2017. Reduksi fitur untuk optimalisasi klasifikasi tumor payudara berdasarkan data citra FNA. *E-Proceedings KNS&I STIKOM Bali* 73–78.
- Witkin, A.P., 1983. Scale-space filtering. Presented at the In Proceedings of the 8th International Joint Conference on Artificial Intelligence, Karlsruhe, Germany, pp. 1019–1023.
- Wong, T.-T., Yeh, P.-Y., 2020. Reliable Accuracy Estimates from k-Fold Cross Validation. *IEEE Transactions on Knowledge and Data Engineering* 32, 1586–1594. <https://doi.org/10.1109/TKDE.2019.2912815>
- Xian, G., 2010. An identification method of malignant and benign liver tumors from ultrasonography based on GLCM texture features and fuzzy SVM. *Expert Systems with Applications* 37, 6737–6741.
- Yang, C., Everitt, J.H., 2010. Mapping three invasive weeds using airborne hyperspectral imagery. *Ecological Informatics, Special Issue on Advances of Ecological Remote Sensing Under Global Change* 5, 429–439. <https://doi.org/10.1016/j.ecoinf.2010.03.002>
- Zulpe, N., Pawar, V., 2012. GLCM textural features for brain tumor classification. *International Journal of Computer Science Issues (IJCSI)* 9, 354.