LudVision
Remote Detection of Exotic Invasive Aquatic Floral Species using Data from a Drone-Mounted Multispectral Sensor

António José Marques Abreu

Dissertação para obtenção do Grau de Mestre em
Engenharia Informática
(2º ciclo de estudos)

Orientador: Prof. Doutor Luís Filipe Barbosa de Almeida Alexandre
Co-orientador: Dr. João Amaral Santos

Covilhã, Junho de 2022.

Acknowledgements

This dissertation is the result of many hours of work, and it would not be possible without the support of many people. This dissertation is dedicated to everyone involved throughout the process of this work.

First, I want to thank my supervisor, Doctor Luis Alexandre, for his constant support, guidance, and expertise throughout the whole process of this research work. I feel truly honored to have been given the opportunity to work with someone that is both demanding, supportive, and passionate about his work. Ever science, you supervised my licentiate degree project, your continuous sharing of knowledge inspired me to pursue the field of AI and made this dissertation's conclusion possible. My deepest gratitude and hope that our paths will cross again.

I would also like to express my gratitude to my co-supervisor João Amaral, and Filippo Basso, for the opportunity they gave me to join their team. I am truly grateful for your trust and support during this project. As Filippo once told me: "Good projects are always the start of something long-lasting". I hope this to be true and that both our friendships and professional relationships are indeed long-lasting.

To my closest family, I do not have enough words to thank you. You always supported me throughout all these years, were always there when I needed the most, and made sure I had everything I needed to pursue my goals and dreams. I am especially grateful for my parents. I know I sometimes gave you a hard time raising me up, especially when I was younger. You always believed in me, and provided me with more than I could ever ask for. You are my cornerstones, and I aspire to one day be like you. I will never be able to pay the debt I owe you, but at least I hope I made you two proud.

No less important, I want to thank all my friends. Some I have known since we were little kids, and others I met during my academic journey. All of you impacted my life in one way or another, making it more cheerful and worth living. A special thanks to my friends and collages from SociaLab for providing such a great working environment and for never refusing to help me. Namely, Daniel Valente and Pedro Brito, for helping me countless times troubleshoot my PC while working remotely.

Finally, I owe a special thanks to Hasty.ai for providing me with a subscription, free of charge, that allowed me to fully access all the features. It made the annotation process easier and faster, allowing me more time to focus on more critical aspects of this work.

Resumo

O sensoriamento remoto é o processo de detetar e monitorizar as características físicas de uma área, medindo à distância a sua radiação refletida e emitida. É amplamente utilizado para monitorizar ecossistemas, principalmente tendo em vista a sua preservação. Há cada vez mais casos de espécies invasoras que afetam o equilíbrio natural dos ecossistemas. As espécies exóticas invasoras têm um impacto crítico quando introduzidas em novos ecossistemas e podem levar à extinção de espécies nativas. Neste estudo, focamo-nos na Ludwigia peploides, considerada pela União Europeia como uma espécie aquática invasora. A sua presença pode ter impactos negativos no ecossistema circundante e nas atividades humanas, como agricultura, pesca e navegação. O nosso objetivo foi desenvolver um método para identificar a presença da espécie. Para isso, usámos imagens capturadas por um sensor multiespectral montado num drone. Devido à falta de conjuntos de dados disponíveis publicamente contendo Ludwigia peploides, tivemos que criar nosso próprio conjunto de dados. Começámos por cuidadosamente estudar todas as opções disponíveis. Primeiro fizemos experiências com imagens de satélite, mas foi impossível identificar a espécie-alvo devido à baixa resolução das imagens. Assim, decidimos usar um sensor multiespectral montado num drone. Infelizmente, devido a limitações orçamentais, não conseguimos adquirir os tipos de equipamentos altamente especializados que são tipicamente usados em sensoriamento remoto. No entanto, estávamos confiantes de que nossa configuração seria suficiente para extrair a assinatura espectral da espécie, e que a alta resolução das nossas imagens comparadas com de satélite, nos permitiria usar modelos de aprendizagem profunda para identificar as espécies.

O uso do drone permitiu uma maior flexibilidade operacional e cobertura de uma grande área. O sensor multiespectral permitiu-nos alavancar as informações de duas bandas adicionais fora do espectro visível. Depois de visitar o local de estudo várias vezes e capturar dados em vários momentos do dia, criámos um conjunto de dados representativo com diferentes condições atmosféricas. Após a captura de dados, procedeu-se às etapas de pré-processamento e anotação para ter um conjunto de dados utilizável. Em etapas posteriores, provámos que é possível extrair dos nossos dados a assinatura espectral da espécie. Esta foi uma conclusão significativa, pois comprovou que de fato é possível diferenciar a assinatura espectral da espécie com equipamentos não tão avançados e especializados quanto os utilizados noutros estudos.

Depois de termos um conjunto de dados, focamo-nos no próximo passo, que foi desenvolver e validar um método que fosse capaz de identificar Ludwigia p. nos nossos dados. Decidimos usar modelos de segmentação semântica para identificar as espécies. Dado que temos apenas duas bandas adicionais em comparação com as imagens RGB tradicionais, não poderíamos abordar o problema como um problema de espectroscopia de sensoriamento remoto padrão. Ao usar modelos de segmentação semântica, podemos aproveitar

não só os recursos desses modelos para reconhecer objetos, mas também a natureza multiespectral de nossos dados. Fundamentalmente, o modelo tem o mesmo comportamento usual, mas tem acesso às informações de duas bandas adicionais.

Começamos por usar um modelo de segmentação semântica estado-da-arte existente, que foi adaptado para lidar com nossos dados. Depois de fazer alguns testes iniciais e estabelecer uma base de comparação, propusemos e implementámos algumas modificações ao modelo existente. O objetivo das modificações foi criar um modelo com menores tempos de treino e melhor desempenho na deteção de Ludwigia p. em altitudes elevadas. O resultado é um novo modelo mais adequado aos nossos dados e aplicação. O nosso modelo é mais rápido no que diz respeito ao tempo de treino, mantendo desempenho semelhante, apresentando mesmo um ligeiro aumento de desempenho em imagens de alta altitude.

Palavras-chave

Sensoriamento Remoto, Ludwigia peploides, Espécies Invasoras Alienígenas, Assinatura Espectral, Satélite, Sensor Multiespectral Montado em Drone, Inteligência Artificial, Segmentação Semântica.

Resumo alargado

O sensoriamento remoto é o processo de detetar e monitorizar as características físicas de uma área, medindo à distância a sua radiação refletida e emitida. É amplamente utilizado para monitorizar ecossistemas, principalmente tendo em vista sua preservação. Há cada vez mais casos de espécies invasoras que afetam o equilíbrio natural dos ecossistemas. As espécies exóticas invasoras têm um impacto crítico quando introduzidas em novos ecossistemas e podem levar à extinção de espécies nativas. Neste estudo, focamo-nos principalmente na Ludwigia peploides, considerada pela União Europeia como uma espécie aquática invasora. Esta planta é caraterizada por formar largos mantos na superfície da água, que impedem a passagem de luz. Tem também uma flor amarela muito caraterística que flore entre o meio da primavera e o início do outono. Durante este período, é muito fácil avistar a presença da espécie em locais afetados. Mais ainda, esta espécie liberta substâncias alelopáticas, como forma de evitar que outras espécies, nomeadamente as nativas, consigam proliferar nas suas imediações. A sua presença pode ter impactos negativos no ecossistema circundante e nas atividades humanas, como agricultura, pesca, navegação e até mesmo na qualidade da água potável. A Ludwigia peploides, foi avistada na Barragem da Touliuca (Zabreia, Portugal) em 2020, e desde então tem-se reproduzido rapidamente, pondo em causa a usabilidade e a qualidade da água da barragem.

O nosso objetivo foi desenvolver um método para identificar a presença da espécie. Para isso, usámos imagens capturadas por um sensor multiespectral montado num drone. Devido à falta de conjuntos de dados disponíveis publicamente contendo Ludwigia peploides, tivemos que criar nosso próprio conjunto de dados. Começámos por estudar cuidadosamente todas as opções disponíveis. Primeiro fizemos experiências com imagens de satélite, mas foi impossível identificar a espécie-alvo devido à baixa resolução das imagens, uma vez que a espécie ainda não ocupa uma área considerável. Assim, decidimos usar um sensor multiespectral montado num drone. Existem vários tipos de drones e de sensores, quer multi quer hiperespectrais que foram concebidos para monitorização de plantas e ecossistemas. Infelizmente, devido a limitações orçamentais, não conseguimos adquirir os equipamentos altamente especializados que são tipicamente usados em sensoriamento remoto. No entanto, estávamos confiantes de que nossa configuração seria suficiente para extrair a assinatura espectral da espécie, e que a alta resolução das nossas imagens comparadas com as de satélite, nos permitiria usar modelos de aprendizagem profunda para identificar a espécie.

O nosso drone é um DJI P4 Multispectral. Trata-se de um drone que tem uma câmara multiespectral já incluída e montada. Esta câmara conta com um sensor RGB tradicional, bem como mais um sensor para cada uma das bandas RGB e dois sensores que captam luz fora do especto visível. Nomeadamente, um sensor RGB, um R (vermelho), um G (verde), um B (azul), um *Red Edge* (Índice RedEdge de Diferença Normalizada) e um *Near-infra red* (perto do espectro infravermelho). Cada um destes sensores capta uma imagem individual, para um total de seis imagens. Uma para cada uma das bandas, que

são monocromáticas e do formato TIFF, e uma imagem tradicional a cores no formato JPEG. O formato TIFF é um formato sem perdas, tipicamente usado em aplicações como o sensoriamento remoto, onde é essencial não haver perda de informação. O uso do drone permitiu uma maior flexibilidade operacional e cobertura de uma grande área. O sensor multiespectral permitiu-nos alavancar as informações de duas bandas adicionais fora do espectro visível. Depois de visitar o local de estudo várias vezes e capturar dados em vários momentos do dia, criámos um conjunto de dados representativo com diferentes condições atmosféricas.

Após a captura de dados, procedeu-se às etapas de pré-processamento e anotação para ter um conjunto de dados utilizável. No processo de pré-processamento, foi necessário proceder ao alinhamento de cada uma das imagens, referente a cada uma das bandas. Uma vez que cada imagem é captada por um sensor diferente, devido ao seu posicionamento (numa disposição matricial de três por dois) e ligeiras diferenças nas propriedades das lentes devido a tolerâncias no processo de fabrico, cada uma das imagens tem uma perspetiva singular comparada com as restantes. Como tal, para criar uma imagem que contenha a informação de todas as bandas, não basta simplesmente juntar as bandas. É necessário primeiro calcular as relações homográficas das imagens e fazer o seu respetivo alinhamento, de forma a que os pixeis das várias imagens alinhem. Só após proceder ao alinhamento das imagens, é possível fazer a sua junção e criar a imagem final. Para fazer a anotação, usámos ferramentas de anotação disponíveis *online* para o efeito. Após completados estes passos essenciais, foi possível a criação de um conjunto de dados.

Em etapas posteriores, provámos que é possível extrair dos nossos dados a assinatura espectral da espécie. Para tal, extraímos a informação espectral de vários pontos da imagem (correspondendo a vários objetos). Após criar as assinaturas com a informação extraída das diversas bandas procedemos à sua comparação. Esta foi uma conclusão significativa, pois comprovou que de fato é possível diferenciar a assinatura espectral da espécie com equipamentos não tão avançados e especializados quanto os utilizados noutros estudos.

Depois de termos um conjunto de dados, focámo-nos no próximo passo, que foi desenvolver e validar um método que fosse capaz de identificar Ludwigia p nos nossos dados. Decidimos usar modelos de segmentação semântica para identificar as espécies. Dado que temos apenas duas bandas adicionais em comparação com as imagens RGB tradicionais, não poderíamos abordar o problema como um problema de espectroscopia de sensoriamento remoto padrão. Ao usar modelos de segmentação semântica, podemos aproveitar não só os recursos desses modelos para reconhecer objetos, mas também a natureza multiespectral de nossos dados. Fundamentalmente, o modelo tem o mesmo comportamento usual, mas tem acesso às informações de duas bandas adicionais.

Começámos por usar um modelo de segmentação semântica estado-da-arte existente, que foi adaptado para lidar com nossos dados. Para tal, foi necessário alterar as camadas de

entrada e algumas subsequentes, uma vez que as nossas imagens têm duas bandas adicionais, quando comparadas com as RGB. Primeiramente realizámos uma série de testes de forma a avaliar o desempenho do modelo original no nosso conjunto de dados. Estes resultados foram usados como uma base de comparação, para validar o nosso modelo. Apesar dos bons resultados do modelo inicial, identificámos algumas áreas que podiam ser melhoradas. Nomeadamente, reduzir os tempos de treino e melhorar o desempenho do modelo em imagens captadas a altitudes superiores. Propusemos e implementámos algumas modificações ao modelo existente e realizamos testes iniciais para as validar. Posteriormente, todos os testes realizados no modelo original foram repetidos no novo modelo, exatamente nos mesmos conjuntos de treino, teste e validação. O resultado é um novo modelo mais adequado aos nossos dados e aplicação. O nosso modelo é mais rápido (cerca de duas vezes mais rápido) no que diz respeito ao tempo de treino, mantendo desempenho semelhante, apresentando mesmo um ligeiro aumento de desempenho em imagens de alta altitude.

Abstract

Remote sensing is the process of detecting and monitoring the physical characteristics of an area by measuring it's reflected and emitted radiation at a distance. It is being broadly used to monitor ecosystems, mainly for their preservation. There have been ever-growing reports of invasive species affecting the natural balance of ecosystems. Exotic invasive species have a critical impact when introduced into new ecosystems and may lead to the extinction of native species. In this study, we focus on Ludwigia peploides, considered by the European Union as an aquatic invasive species. Its presence can have negative impacts on the surrounding ecosystem and human activities such as agriculture, fishing, and navigation. Our goal was to develop a method to identify the presence of the species. To achieve this, we used images collected by a drone-mounted multispectral sensor. Due to the lack of publicly available data sets containing Ludwigia peploides, we had to create our own data set. We started by carefully studying all the available options. We first experimented with satellite images, but it was impossible to identify the targeted species due to their low resolution. Thus, we decided to use a drone-mounted multispectral sensor. Unfortunately, due to budget limitations, we could not acquire the highly specialized types of equipment that is more commonly used in remote sensing. However, we were confident that our setup would be enough to extract the species' spectral signature, and that the higher resolution compared to satellites would allow us to use deep learning models to identify the species.

The use of the drone allowed for better operational flexibility and to cover a large area. The multispectral sensor allowed us to leverage the information of two additional bands outside the visible spectrum. After visiting the study site multiple times and capturing data at various times of the day, we created a representative data set with different atmospheric conditions. After the data collection, we proceeded to the pre-processing and annotation steps to have a usable data set. In later stages, we proved that extracting the specie's spectral signature from our data set is possible. This was a significant conclusion, as it proved that it is indeed possible to differentiate the species' spectral signature with equipment that is not as advanced and specialized as the ones used in other studies.

After having a data set, we focused on the next step, which was to develop and validate a method that would be able to identify Ludwigia p on our data. We decided on using semantic segmentation models to identify the species. Given that we only have two additional bands compared to traditional RGB images, we could not approach the problem as a standard remote sensing spectroscopy problem. By using semantic segmentation models, we can leverage both the capabilities of these models to recognize objects and the multispectral nature of our data. Fundamentally, the model has the same behavior as usual but has access to the information of two additional bands.

We started by using an existing state-of-the-art semantic segmentation model adapted to handle our data. After doing some initial tests and establishing a baseline, we proposed and implemented some changes to the existing model. The goal of the modifications was to create a model with lower training times and better performance in detecting Ludwigia p. at high altitudes. The result is a new model better suited to our data and application. Our model is faster when it comes to training time while maintaining similar performance and has a slight performance increase in high-altitude images.

Keywords

Remote Sensing, Ludwigia peploides, Invasive Alien Species, Spectral Signature, Satellite, Drone-Mounted Multispectral Sensor, Artificial Intelligence, Semantic Segmentation.

List of Figures

List of Tables

Acronyms and Abbreviations

AI	Artificial Intelligence
ASPP	Atrous Spatial Pyramid Pooling
AVIRIS	Airborne Visible and InfraRed Imaging
AVIRIS-ng	Airborne Visible and InfraRed Imaging Spectrometer — next generation
CD	Change Detection
CNN	Convolutional Neural Network
CSTARS	Center for Southeastern Tropical Advanced Remote Sensing
DCNN	Deep Convolutional Neural Network
DJI	Da-Jiang Innovations
DN	Digital Number
DT	Decision Tree
FA	Factor Analysis
FAA	Federal Aviation Administration
FoV	Field of View
GCL	Global Convolutional Long Short-term Memory
GCP	Ground Control Point
GJAM	Global Joint Attention Mechanism
GPS	Global Positioning System
HSI	Hyperspectral Image
HyMap	Airborne Hyperspectral Imaging
IAS	Invasive Alien Species
InSb	Indium Antimonide
IoU	Intersection over Union
JPL	Jet Propulsion Laboratory (Pasadena, California, USA)
MAV	Manned Aerial Vehicle
ML	Machine Learning

MNF	Minimum Noise Fraction
MSRA	Microsoft Research Asia
OCR	Object Contextual Representation
OSCD	Optimal Scale Change Detection
PCA	Principal Component Analysis
RF	Random Forest
RNN	Recurrent Neural Network
ROSIS	Reflective Optics System Imaging Spectrometer
RS	Remote Sensing
SSDGL	Spectral-spatial Dependent Global Learning
SVM	Support Vector Machine
TIFF	Tagged Image File Format
UAV	Unmanned Aerial Vehicle

Chapter 1

Introduction

1.1 Motivation and Objectives

Exotic invasive species have a critical impact when introduced in new ecosystems, and there is a growing global concern due to their negative ecological and economic ramifications. The introduction of invasive plants often results in the extinction of native plants and a reduction in biodiversity, either by outcompeting or hybridizing with native species [10]. This is especially the case in aquatic environments and for aquatic plants [22]. Under most scenarios, invasive plants arrive without their co-evolved competitors or parasites, allowing them to spread rapidly, replacing native plants without assuming their ecological roles.
Invasives in wetlands have many adverse effects within and across trophic levels and greatly reduce biodiversity [37, 48]. Many invasives may directly compete with other species by secreting allelopathic chemicals that reduce germination and seedling survival, or by changing light accessibility [21, 37, 46]. Invasives may also significantly impact invertebrate distribution, diversity, and abundance; induce anoxic conditions detrimental to fish and other aquatic life [21, 38]; and act as barriers for fish movement [46, 48]. They also reduce open water habitat for water birds and other wildlife [48].
Plant invasions have been shown to modify ecosystem processes such as nutrient availability, nutrient cycling, soil chemistry, water tables, hydrology, food waste, and habitats [31]. Management of these potentially detrimental impacts is complicated by changes in climate and intensified by increases in invasion frequency due to globalization. Human-mediated introductions of invasive plants are most common and tend to be more rapid, increasing pressure and exacerbating the threat of the economic and environmental damages associated with invasive plants.

Remote Sensing (RS) image analysis is increasingly being used as a tool for mapping invasive plant species. The resulting distribution maps can be used to target management of early infestations and to model future invasion risks [11, 10]. Remote identification of invasive plants based on differences in spectral signatures is the most common approach, typically using hyperspectral data [26]. Advances in Unmanned Aerial Vehicles (UAVs) and sensor miniaturization are enabling higher spatial resolution species mapping, which is promising for early detection of invasions before they spread over larger areas [9].

The LudVision project aims to control the spread of Ludwigia peploides, specifically in the Reservoir of the Toulica Dam (Zebreira, Portugal), located in the hydrographic basin

of the Aravil river, a tributary of the Tagus, where it was detected in 2020. Ludwigia peploides is a species natural to South America that invades rivers, ponds, and rice fields. It can grow in deep waters, as a fully or partially submerged plant, and form floating mantles. When this happens, it prevents the entry of light affecting submerged species and blocking the water lines, affecting navigation, fishing, and recreational use. It competes for space by eliminating native species and producing substances that inhibit the germination and growth of other species. It reproduces vegetatively through stem fragmentation but also through seeds. It is an invasive species that raises concern at the European level (EU Reg. 1143 [3]) and is included in the Portuguese legislation list of invasive species (DL 92/2019 of 10/07 [1]).

As such, this thesis work aims to develop a system for the remote detection of the Ludwigia peploides. To do so, we propose using aerial data captured with a setup that is cheaper and simpler than the ones used in related work and use semantic segmentation models that will be modified to handle our data.

1.2 Main Contributions

The main contribution of our research work is a new approach for the detection of Ludwigia, using multispectral images captured by a drone-mounted sensor. This approach can be subdivided in four parts: (1) creation of a new data set composed of multispectral images. This data set focuses on the Invasive Alien Species (IAS) Ludwigia peploides present in the Toulica Dam. The captured images, where taken at different days, times of the day, atmospheric conditions, and altitudes. (2) making the data set, publicly available for other researchers; (3) use of a general application sensor and platform, instead of the highly specialized and expensive equipment traditionally used in RS applications; (4) the proposal of a new semantic segmentation method that can handle multispectral data and successfully detect Ludwigia p.
Complementary to this work, an article [5] was written and sent to the Journal Remote Sensing of Environment, published by Elsevier, where it awaits review.

1.3 Thesis Organization

In order to provide an intuitive reading experience, this document is divided in the following chapters:

1. **Introduction:** provides a general overview of this work's motivations and objectives, as well as, the structure by which this document is organized.
2. **Preliminary Concepts:** introduces some fundamental concepts and techniques that support our thesis work.
3. **Related Work:** presents the work done by other authors, that will be used as baselines for this thesis.
4. **Data Sets:** showcases related data sets, available platforms and sensors for data collection, and our LudVision dataset.
5. **A New Method for Detection of Ludwigia in Multispectral Images:** contains the experiments performed to validate our new proposed method, with accompanying results and discussions.
6. **Conclusions and Future Work:** concludes the present document with both a review of the work and its potential improvements in the future.

Chapter 2

Preliminary Concepts

In this chapter, we introduce some fundamental concepts and techniques related to RS, classification methods, semantic segmentation and spectral image classification. Section 2.1 covers RS image analysis, as it is increasingly used for mapping invasive plant species. It allows for the creation of distribution maps that can be used to target specific species and assess the evolution of infestations. RS of IAS relies mainly on the differences in spectral signatures of the different plants and the surrounding environment, allowing for the differentiation and identification of targeted species. Section 2.2 gives an overview of the most common pre-processing operations carried out in RS. Section 2.3 introduces some of the most common ways to assess the accuracy of the generated maps. This is important to evaluate and compare the performance of the algorithms. Section 2.5 analyzes some of the classification models commonly used in RS. Section 2.6 covers state-of-the-art deep learning models used in semantic segmentation. Lastly, section 2.7 presents some models specifically designed for spectral image classification.

2.1 Remote Sensing

RS has been the go-to tool for IAS mapping, especially when it comes to plants, due to its ability to provide synoptic views over large geographical extents. This is an advantage over the more traditional field surveys, which are often limited to small and accessible areas [10, 11, 26]. Improvements in UAVs and sensor technology offer the potential to fill the gap between field surveys and manned flights. UAVs provide imagery with a smaller footprint than manned flights but larger than field surveys. Plus, they have high operation flexibility and low cost, which allows for on-demand launches. This allows more frequent acquisitions that enable: (1) better characterization of both phenological stages and differences in phenology between species, (2) sampling during specific and relevant events such as floods or herbicide treatments [10].
With current technological advancements and the existence of multi and hyperspectral sensors [55], it is possible to distinguish species with great accuracy, even in the same functional groups [9, 31, 49]. The advancement in sensor technology, coupled with advances in image processing and Machine Learning (ML) algorithms, provide accurate and repeatable RS measurements over time, that also provide consistent monitoring records to support control efforts.

Three factors make mapping IAS using RS viable [10]). First, IAS usually grow as large homogeneous patches and tend to have unique growing patterns. Thus, it is somewhat easy to train a classifier that can recognize it. For example, in the case of Ludwigia peploides (but also in other IAS), we can have two scenarios: (1) it is the only IAS present in the body of water. Given that it also grows as large patches, the mapping is as easy as separating green vegetation from the surrounding water, (2) there are other IAS present competing with Ludwigia peploides. However, the patches are well identifiable, dense, and usually not mixed. This is due to allelopathic activities that some IAS use (as is the case for Ludwigia p.), like poisons that discourage other plants from mixing into the patch. Second, when the IAS has unique phenology compared to the surrounding native vegetation, allowing for easier differentiation during some parts of the year. One example of this is during the IAS flowering period. If it has a distinct and unique flower or even if it blooms at different periods from the native species, it makes it easier to identify the IAS. The only constraint with using phenology to identify an IAS is that the data needs to be collected in specific periods. If the period is short or still a couple of months away, it can delay the data collection procedures. In the case of Ludwigia peploides, it is not a big problem because the flowering period lasts from April to November.
Third, the target IAS has unique chemistry or bio-physiology. While two plants can have very similar characteristics, one invasive plant, usually has a very distinctive biophysiology, as it tends to be more resilient to climatic conditions and other factors than native species [44], thus making the IAS easily identifiable during most of the year. This allows for plant identification and differentiation, even when the IAS is outside its phenological relevant period. For example, Khanna et al. [33] successfully differentiated water hyacinth from other co-occurring floating aquatic macrophytes using differences in canopy water content, since water hyacinth has a higher plant-water content than co-occurring species water primrose (Ludwigia peploides) and water pennywort (Hydrocotyle ranunculoides). This requires spectral rich data that can be collected using hyperspectral sensors.

The three abovementioned requirements are matched with three domains of RS data: spatial, temporal and spectral [10]. Generally speaking, and as the name suggests, hyperspectral imagery is rich in data in the spectral domain, aerial imagery from both Manned Aerial Vehicles (MAVs) and UAVs mounted sensors are rich in the spatial domain and satellite imagery in the time domain. Each one of these platforms and sensors has its trade-offs between the domains. Thus is essential to find a balance between them for each targeted IAS. Each species is different, and so is their environment, presenting new challenges and opportunities for IAS mapping using RS. Regardless of the target IAS and habitat, the general process of detecting and mapping IAS remains the same and consists of the following steps [10]:

1. *Identify the target species and area*: How and which IAS is affecting biodiversity, native ecosystems, or economic functions. What do we know about the target IAS (e.g., spectral characteristics, phenology, ecosystem function, habitat requirements);

2. *Determine the appropriate platform/sensor and identify/collect supplementary data based on species and habitat knowledge*: Once the target species is identified, we need to detect it. The detection of species can be achieved in two ways: directly or indirectly. Direct detection uses spectral data and derived products from imagery. Indirect detection utilizes the ecological relationships between species and their environment to predict distribution. Because each species and habitat is different, exploitable differences can exist in the temporal, spatial, or spectral domains. The temporal domain consists of data collection timing and revisiting timing. The spatial domain consists of pixel size and overall geographic coverage, and the spectral domain consists of the number of wavelengths, the position, and bandwidth of wavelengths measured, and the spectral range of the sensor at which radiance can be measured reliably;

3. *Enhance data and model/classify*: This is the step where either supervised or unsupervised data is fed to a model. The model then uses the data to train itself in order to be able to detect the targeted IAS. Methods to enhance spectral data include: Principal Component Analysis (PCA), Minimum Noise Fraction (MNF) and Factor Analysis (FA). PCA is a linear transformation method that maximizes the variance of the data. When applied to a Hyperspectral Image (HSI) it produces a series of components that correspond to linear combinations of the original bands, aligned to represent the variation within the original data set. The first component is the plane responsible for the most variation. This allows for determining the most significant characteristics within an image, related to the problem's classes. MNF rescales the noise in the data (a process called noise whitening), enabling the analyst to eliminate bands containing too much sensor noise and leaving only coherent image data [10]. FA approaches data reduction in a fundamentally different way than PCA. It is a model of the measurement of a latent variable. This latent variable cannot be directly measured with a single variable (e.g., intelligence, social anxiety, soil health). Instead, it is seen through the relationships it causes in a set of Y variables. According to [7], FA reduces the dimensionality of the data set to the number of significant components and describes this space within the full dimensionality of the data through the derived eigenvectors. The corresponding eigenvalues describe the variance along each eigenvector within the system. The information contained within the eigenvalues and eigenvectors can be used to estimate the number of end-member components that influence the data. If independent components vary within a system, the spectra composed of mixtures will be linear combinations of the spectral end-members if spectral mixing is linear.

Commonly used classification techniques include Random Forests (RFs), Support Vector Machines (SVMs), and more recently Convolutional Neural Network (CNN) like 2D CNNs, 3D CNNs, 3D-2D CNNs, etc. Some state-of-the-art spectral classification models like SpectralNET, HybridSN, and SSDGL, have been created specifically to classify spectral data. We will discuss these models in later chapters;

4. *Assess Accuracy*: Once we have trained our model, we must assess its performance and capability to classify the target IAS correctly. Depending on the case, some types of errors may or not be accepted. Typical accuracy metrics for image classification include overall accuracy, user's accuracy, producer's accuracy, and Kappa coefficient [10]. The definitions of these metrics are given in section 2.3. The problem with overall accuracy is that it does not consider the number of validation pixels per class. This can be misleading if the data for each class is not balanced in the validation phase. Because of this, user's and producer's accuracy are often used for assessing classification performance. Lastly, the Kappa coefficient is helpful to compare multiple classification methods using the same data set.

2.1.1 Invasive Alien Species in Aquatic Ecosystems

Each IAS and the respective habitats they are invading are unique. Thus, identifying, mapping, and detecting the targeted IAS using RS is a unique challenge. This is due to different landscape configurations, community composition, canopy structures, climates, habitat characteristics, and plant phenology. For this reason RS, for IAS mapping and detection is divided into more specific ecosystems and case studies. Because the scope of this thesis is the mapping and detection of the species Ludwigia peploides, which is an aquatic IAS, we will be focusing solely on the use of RS in aquatic ecosystems.

The Earth's aquatic ecosystems are one of the most diverse and complex ecosystems. Aquatic ecosystems encompass multiple gradients, such as water intermittency, microtopography and salinity, leading to complex environmental heterogeneity. This diversity poses both challenges and opportunities for the use of RS for aquatic IAS mapping and detection. Plants in aquatic ecosystems can be divided/classified in five functional types that occupy distinct spatial niches along the gradient from land to water and often have similar characteristics: riparian forests with shrubs and trees, emergent reeds and sedges, floating macrophytes, submerged macrophytes and macroalgae, and phytoplankton [10]. Differentiation species among the functional types is achievable with RS. The more difficult task is to differentiate species in the same functional type because they have very similar characteristics and phenology. Before analyzing related work and articles in RS, it is crucial first to discuss the commonly used techniques that allow for plant identification and differentiation: Spectral measurements, Photophysiological Measurements, Textural and Object-Based Differentiation, Phenological Analysis, and Seasonal Dynamics, and Change Detection.

2.1.2 Spectral Measurements

Currently, most studies aim to map invasive species remotely by using the differentiation in spectral signatures. Spectral differentiation works when the target invasive species has at least one unique light reflectance or absorption feature, relative to the surrounding vegetation and environment. This gives the targeted species a unique spectral signature (like the examples in Figure 2.1) that can be used to identify it. In order to register the unique spectral signature, multispectral or preferably hyperspectral imagery needs to be used. As a rule of thumb, hyperspectral imagery is rich in data in the spectral domain, aerial imagery from piloted and unpiloted aircraft in the spatial domain, and satellite imagery in the time domain. These platforms and sensor types have tradeoffs between the three domains and are typically only strong in one. Selecting the best platform/sensor and fusing the collected imagery with appropriate supplementary data results in the best classification maps.

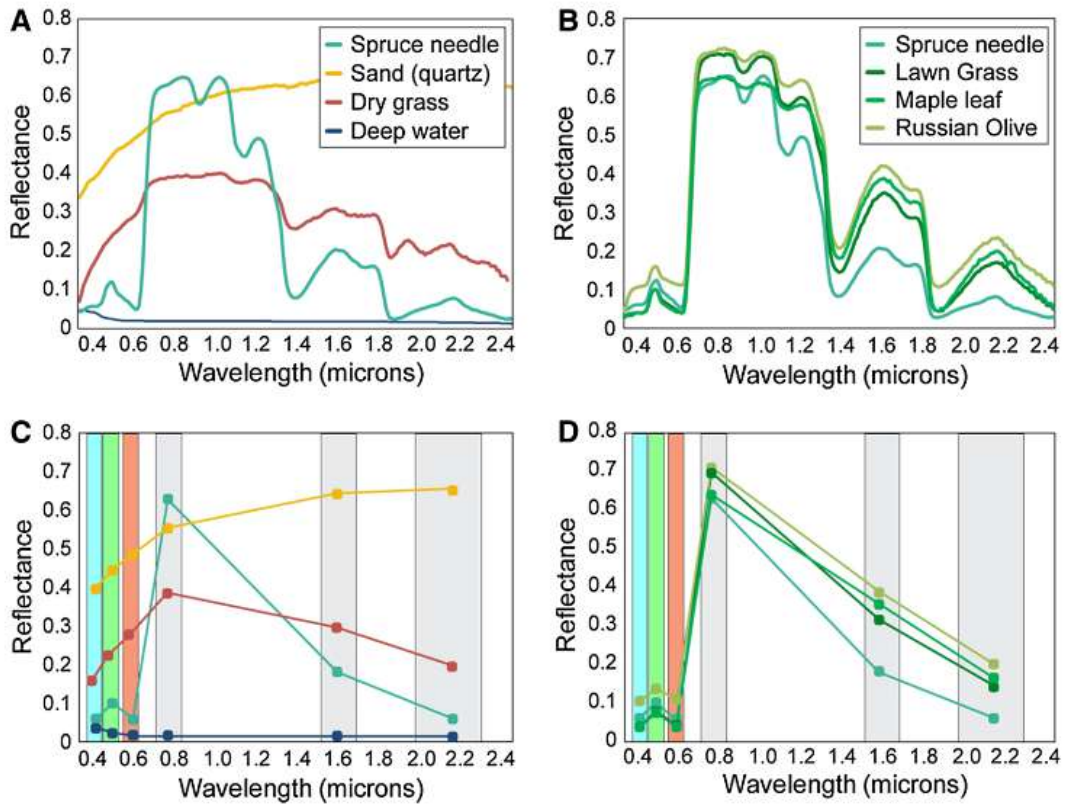


Figure 2.1: Examples of spectral signatures calculated from hyperspectral (A and B) and multispectral (C and D) imagery [11].

2.1.3 Photophysiological Measurements

This technique focuses on measuring the chlorophyll fluorescence emitted from plant leaves. Similar to spectral measurements, the plant identification using this method is possible because chlorophyll fluorescence is slightly different between plants. Chlorophyll fluorescence parameters are measured using chlorophyll fluorometers that are designed to measure variable fluorescence of photosystem II. By identifying the levels of chlorophyll fluorescence of the targeted plant, we can use that information to identify the invasive species remotely.

2.1.4 Textural and Object-Based Differentiation

Textural and object-based differentiation, like the name implies, is the technique of identifying patterns within adjacent pixel neighborhoods. Textural analysis recognizes a particular pattern and direction among a group of pixels. Object-based analysis is similar in some ways, as it focuses on identifying a particular object from the surrounding pixels. The main difference between the two is that the target object must be larger than the pixel size to be effectively identified.

2.1.5 Phenological Analysis and Seasonal Dynamics

We can use phenological detection if the invasive species has a different seasonal or interannual growth pattern different from the native species. In most cases, the invasive plant has some advantage over the native species. Hence they usually outgrow native populations. The invasive species might bloom earlier or later than the native species or has a particular flower. This allows for remote detection using the distinct phenological patterns. This technique requires a repeated time-series image acquisition of the study site. By comparing the evolution of the different patterns, it is possible to identify the invasive species.

2.1.6 Change Detection

This method works similarly to the previous one, but it implies that we have long-term data of the study area. The data must have a consistent mapping approach and use the same, or at least similar, imagery. By going back in time in the long-term data, we can observe the evolution of an invasive plant from its early stages. If we gather enough long-term information about the targeted species, we can compare the growth pattern of our study site with growth patterns of the same species in other places or even environments.

2.2 Image Pre-processing in Remote Sensing

Most of the time, the data collected with remote sensing techniques is delivered without any pre-processing. Pre-processing operations such as image restoration and rectification, are intended to correct for sensor and platform-specific radiometric and geometric distortions of data. Radiometric corrections may be necessary due to variations in scene illumination and viewing geometry, atmospheric conditions, and sensor noise and response. Each of these will vary depending on the specific sensor and platform used to acquire the data and the conditions during data acquisition. Also, it may be desirable to convert and/or calibrate the data to known (absolute) radiation or reflectance units to facilitate comparison between data. In this section, we will cover two of the most common pre-processing operations in RS: geometric correction and atmospheric correction.

2.2.1 Geometric Correction

Geometric correction (also known as geo-correction) transforms the X and Y dimensions of a remotely sensed image so that original distortions are eliminated or at least minimized and the X and Y dimensions of the output image correspond to a chosen geometric reference system. Geo-correction is necessary because satellite flight paths typically do not align with most geographic reference systems' true north and the grid orientation. Geometric correction is usually done in the following scenarios:

1. Mosaic together two or more remotely sensed images into a single, combined image;
2. Compare two or more remotely sensed images of the same area from different times;
3. Locate points and features of interest on the geometrically corrected image;
4. Accurately calculate distance and area from the geo-corrected image.

One way of geo-correcting an image is using Ground Control Points (GCPs). These correspond to locations that can be precisely identified in the remotely sensed image and the target geographic reference system (ideally, at least 20 points are needed). Once the points have been chosen, we record their respective X and Y coordinates in the target geographic reference system. Finally, we use equations 2.1 and 2.2 to transform input X and Y coordinates to the desired output reference system. The coefficients a and b in these equations should be solved, using the x and y of the coordinates in the geometrically corrected data, and the u and v of the uncorrected satellite data. The coordinate pairs x, y and u, v are the GCPs.

$$u = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2 \quad (2.1)$$

$$v = b_0 + b_1x + b_2y + b_3xy + b_4x^2 + b_5y^2 \quad (2.2)$$

2.2.2 Atmospheric Correction

The atmosphere affects the spatial and spectral distribution of the electromagnetic radiation originating from the sun before it reaches the Earth's surface, and it also attenuates the subsequently reflected energy recorded by a satellite sensor. Gas absorptions molecule and aerosol scattering are examples of atmospheric processes that influence incident and reflected radiation. Atmospheric correction removes the scattering and absorption effects from the atmosphere to obtain the surface reflectance characterizing (surface properties). Atmospheric correction is needed mainly for two reasons: atmospheric transmittance (the proportion of ground radiance that reaches the sensor) and atmospheric path radiance (reflection from atmospheric particulates results in an additional radiance that did not originate from the Earth's surface).
While for some applications, atmospheric correction may not be necessary, for others, it is essential (e.g., performing a time-series analysis in crop growth). Thus, the information on the atmospheric conditions present at the time/period of image acquisition needs to be recorded for the applications that require atmospheric image correction.
During the atmospheric correction, the image pixel values (known as Digital Numbers (DNs)) are converted to a physically interpretable measure, often referred to and interpreted as surface reflectance. This conversion is done in two steps: first is radiometric calibration, which involves the conversion of DNs to radiance, and then to top-of-atmosphere-radiance. The obtained values can be interpreted as radiance observable just outside of the Earth's atmosphere; their derivation from the DNs can generally be done with just the metadata that is delivered with the image. In the second step, the top-of-atmosphere reflectance is converted to surface reflectance (also known as bottom-of-atmosphere reflectance, top-of-canopy reflectance, or vegetation studies). Top-of-canopy reflectance can be understood as reflectance as would be measured from just above the vegetation. This phase requires knowledge of atmospheric conditions present during the image acquisition time frame. After these two steps, we have an atmospherically corrected image.

2.3 Map Accuracy Assessment

In remote sensing, it is vital to assess the accuracy of the generated maps. This section, presents the main methods and metrics to evaluate the accuracy of the classification. The most common form of expressing classification accuracy is the error matrix (confusion matrix), like the example in figure 2.2. The error matrix compares, on a class-by-class basis, the relationship between known reference data (ground truth) and the corresponding results of the classification procedure, allowing to calculate the following accuracy metrics:

- Overall Accuracy;
- Errors of Omission;
- Errors of Commission;
- Producer's Accuracy;
- User's Accuracy;
- Kappa Coefficient.

		Reference Data			
		Water	Forest	Urban	Total
Classified Data	Water	21	6	0	**27**
	Forest	5	31	1	**37**
	Urban	7	2	22	**31**
	Total	**33**	**39**	**23**	***95***

Figure 2.2: Example of a confusion matrix [4].

2.3.1 Overall Accuracy

Overall accuracy tells us what proportion of the reference sites were mapped correctly. The overall accuracy is usually expressed as a percent, with 100% accuracy being a perfect classification where the whole reference site was classified correctly. Overall accuracy is the easiest to calculate and understand but ultimately only provides the map user and producer with basic accuracy information.
The diagonal elements represent the areas that were correctly classified. To calculate the overall accuracy, we add the number of correctly classified sites and divide it by the total number of reference sites.

2.3.2 Errors of Omission

Errors of omission (sometimes also referred to as a Type I error) refer to reference sites left out (or omitted) from the correct class in the classified map. An error of omission in one category will be counted as an error in commission in another category. Omission errors are calculated by reviewing the reference sites for incorrect classifications. This is

done by going down the columns for each class and adding together the incorrect classifications and dividing them by the total number of reference sites for each class. A separate omission error is generally calculated for each class. This will allow us to evaluate each class's classification accuracy and error.

2.3.3 Errors of Commission

Errors of omission concern the classified results. These refer to sites that are classified as reference sites but were left out (or omitted) from the correct class in the classified map. Commission errors are calculated by reviewing the classified sites for incorrect classifications. This is done by going across the rows for each class and adding together the incorrect classifications. Then divide them by the total number of classified sites for each class.

2.3.4 Producer's Accuracy

Producer's accuracy is the map accuracy from the point of view of the mapmaker (the producer). This is, how often the real features on the ground are correctly shown on the classified map. Alternatively, the probability that a certain land cover of an area on the ground is classified as such. The producer's accuracy is the complement of the omission error, $Producer's\ Accuracy = 100\% - Omission\ Error$. It is also the number of reference sites classified accurately, divided by the total number of reference sites for that class.

2.3.5 User's Accuracy

The user's accuracy is the accuracy from the point of view of a map user. The user's accuracy essentially tells how often the class on the map will be present on the ground. This is referred to as reliability. The user's accuracy is the complement of the commission error, $User's\ Accuracy = 100\% - Commission\ Error$. The user's accuracy is calculated by taking the total number of correct classifications for a particular class and dividing it by the row total.

2.3.6 Kappa Coefficient

The kappa coefficient or Cohen's kappa statistic, is generated from a statistical test to evaluate the accuracy of a classification. Kappa essentially evaluate how well the classification performed as compared to just randomly assigning values (e.g., did the classification do better than random). The kappa coefficient can range from -1 to 1. A value of 0 (zero) indicated that the classification is no better than a random classification. A negative number indicates the classification is significantly worse than random. A value close to 1 indicates that the classification is significantly better than random.

2.4 Semantic Segmentation Evaluation Metrics

Given that we plan on using semantic segmentation methods to identify the presence of Ludwigia p., we will present the most relevant semantic segmentation metrics and how they are calculated. This will allow us to better understand and compare the results of our model.

2.4.1 Intersection Over Union

Intersection over Union (IoU) is a metric that determines the extent of overlap between two areas. In the case of semantic segmentation, it is used to determine the overlap between the ground truth and the model's predictions. This metric ranges from 0 to 1 (or 0 to 100%), where 1 means a perfect overlap. For multi-class segmentation, we can also calculate the class and mean IoU. As the names suggest, class IoU is calculated by evaluating the overlap for each class, and the mean IoU is the mean value of all classes.
This metric can be calculated using equation 2.3

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (2.3)$$

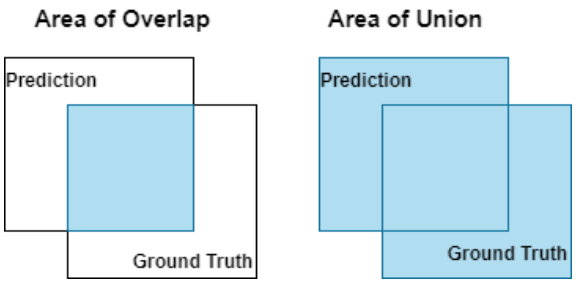


Figure 2.3: Illustration of the overlap and union areas.

2.4.2 Pixel Accuracy

This is probably the most intuitive metric to calculate and understand. It is the percentage of correctly classified pixels in the image. It is the same as the overall accuracy covered in 2.3.1.
Despite its intuitiveness, this metric can be misleading and should never be the only metric used to evaluate a model's performance. This is especially the case for class-imbalanced datasets (datasets with an uneven number of instances per class). Let us consider the following example (for simplicity, consider a dataset with just one image): That image has 95% of pixels corresponding to class 1. If the model classified all pixels from the image as being from class 1, using the pixel accuracy metric, it would have 95% accuracy. This would seem like a good result, but the model failed to classify the pixels of the other classes correctly. Thus, this metric is only representative for balanced datasets.

2.5 Classification Algorithms

The main idea of classification algorithms is very simple. Use a training dataset to get better boundary conditions which could be used to determine each target class. Once the boundary conditions are determined, the next task is to predict the target class. This section analyses some of the classical classification algorithms. Note, that the covered algorithms, are some of the most commonly used in RS applications.

2.5.1 Decision Trees

A Decision Tree (DT) is a type of machine learning model used to solve regression and classification problems. DTs are a non-parametric supervised learning method used for classification and regression. DTs learn from data to approximate with a set of if-then-else decision rules. The deeper the tree, the more complex the decision rules, and the fitter the model. The classification or regression models are built in the form of a tree structure, hence the name. The data is broken down into smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes.

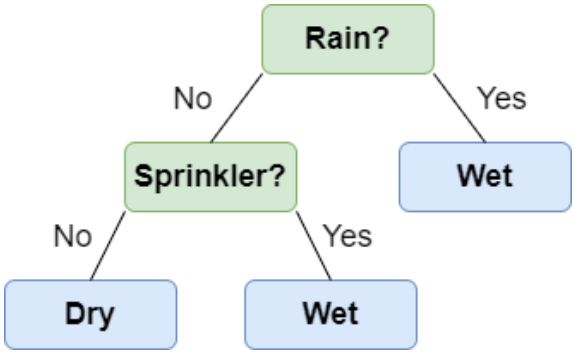


Figure 2.4: Visual representation of a DT.

2.5.2 Random Forests

RFs are also a type of machine learning model used to solve regression and classification problems. They are based on ensemble learning (combines multiple classifiers to provide a solution to more complex problems). A RF is an automated algorithm that builds many decision trees. It establishes an output based on the predictions of the decision trees. It takes the average or mean of the output of the various trees. A RF eliminates the limitations of a DT algorithm. It reduces the overfitting of datasets and increases precision. Increasing the number of trees usually reflects an improvement of the RF algorithm results, but it comes with an exponentially added computational cost. So, there is a need for a balance between the desired performance of the algorithm and the available computational power plus the time constraints.

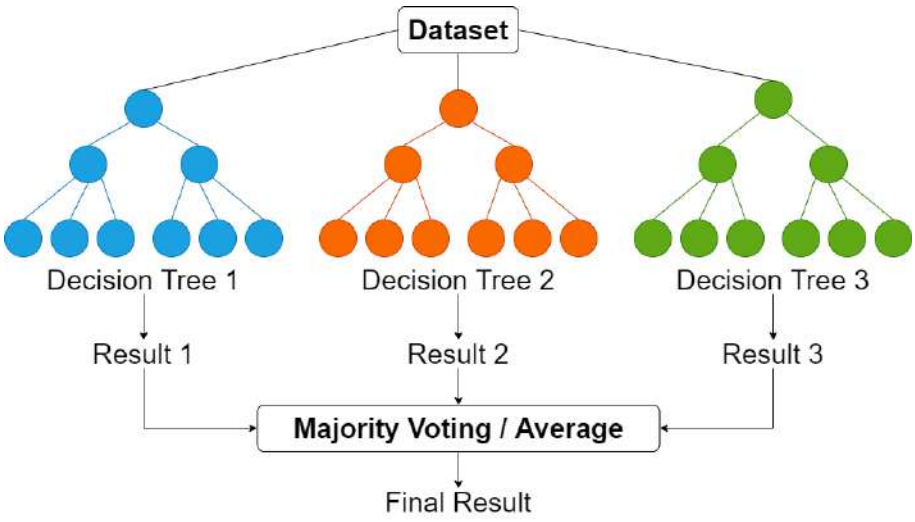


Figure 2.5: Visual representation of a RF.

2.5.3 Support Vector Machines

An SVM is a linear model for classification and regression problems. It can solve linear and non-linear problems and works well for many practical problems. The way SVMs work is simple: the algorithm creates a line or a hyperplane which separates the data into classes. The distance between the hyperplane and the nearest data point from either set is the margin. The goal is to choose a hyperplane with the greatest possible margin between the hyperplane and any point within the training set, giving a greater chance of new data being classified correctly. The process of training an SVM, is finding the ideal parameters that define the best possible hyperplane.

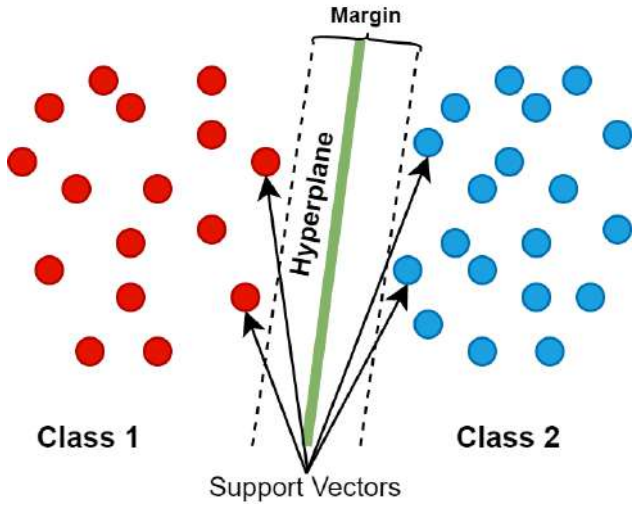


Figure 2.6: Visual representation of a SVM.

2.6 Semantic Segmentation

The semantic segmentation task consists of classifying each pixel of an image. Prior to deep and machine learning, feature extraction was a tedious and time-consuming step that needed to be made by a human. Nowadays, deep learning has evolved to a stage where in some instances, it outperforms humans. It uses images as data to train itself and extract the most relevant features of any given object. This allows the models to then generalize what they learned on the training data to identify objects in new data instances. This section, presents some of the state-of-the-art semantic segmentation models.

2.6.1 DeepLabV3

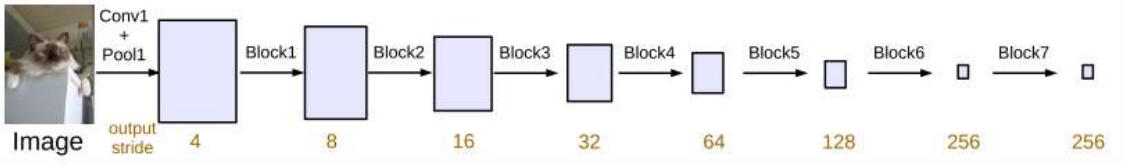
DeepLabV3 [17] is an improvement over the previous DeepLab [14] versions. The authors consider that there are two challenges in applying Deep Convolutional Neural Networks (DCNNs) to semantic segmentation. The first is the reduced feature resolution caused by the consecutive pooling operations or convolution striding. These operations allow for the DCNNs to learn abstract feature representations. However, the invariance to local image transformation may impede dense prediction tasks, where detailed spatial information is needed. To overcome the issue, they propose using atrous convolutions (also known as dilated convolution), which will be discussed later.
The second challenge, is the existence of objects at multiple scales. Several methods have been proposed by the authors to handle the problem.

2.6.1.1 Atrous Convolution for Dense Feature Extraction

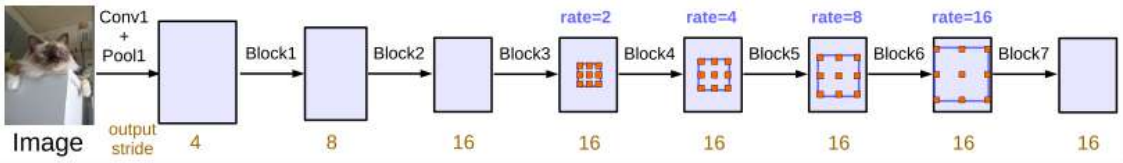
Fully convolutional DCNNs, have shown to be effective in semantic segmentation. However, the repeated combination of max-pooling and striding at consecutive layers of these networks significantly reduces the spatial resolution of the resulting feature maps. Usually, deconvolutional layers have been used to recover some spatial resolution. However, the authors propose the use of atrous convolutions. These convolutions extract denser feature maps by removing the downsampling operations from the last few layers and upsampling the corresponding filter kernels. This is equivalent to inserting holes between filter weights [15].

2.6.1.2 Going Deeper with Atrous Convolution

The proposed model's motivation is that the striding makes it easy to capture long-range information in the deeper blocks. Nevertheless, the consecutive use of striding is harmful for semantic segmentation since detail information is decimated, and thus the authors apply atrous convolution with rates determined by the desired output stride value. An illustration of applying atrous convolution with rated determined by the desired output stride value (in this case $output_stride = 16$) can be seen in figure 2.7.



(a) Going deeper without atrous convolution.



(b) Going deeper with atrous convolution. Atrous convolution with $rate > 1$ is applied after block3 when $output\_stride = 16$.

Figure 2.7: Cascaded modules with (b) and without (a) atrous convolution [14].

2.6.1.3 Atrous Spatial Pyramid Pooling

Inspired by the success of spatial pyramid pooling at effectively resampling features at different scales for accurately and efficiently classifying regions of an arbitrary scale, the authors revisited the Atrous Spatial Pyramid Pooling (ASPP) proposed in [14] and included batch normalization with ASPP. Despite ASPP being able to capture multi-scale information effectively, as the sampling rate becomes larger, the number of valid filter weights becomes smaller.

2.6.2 DeepLabV3+

DeepLabV3+ [17], extends DeepLabV3 [17] by adding a simple yet effective decoder module to refine the segmentation results, especially along object boundaries. The authors, also further explore the Xception model and apply the depthwise separable convolution to both ASPP and decoder modules, resulting in a faster and stronger encoder-decoder network.

2.6.2.1 Proposed Decoder

The encoder features from the previous DeepLabV3 [15] are usually computed with *output stride* $= 16$. In the experiments conducted in [15], the features are bilinearly upsampled by a factor of 16, which is considered naive. The problem with this decoder is that it may not successfully recover object segmentation details. Thus the authors propose a new, simple yet effective decoder module illustrated in figure 2.8. The encoder features are first bilinearly upsampled by a factor of 4. Then they are concatenated with the corresponding low-level features from the network backbone that have the same spatial resolution. Then, another 1×1 convolution is applied to the low-level features to reduce the number of channels. This is done to prevent the low-level features (that usually have more channels) from outweighing the importance of the rich encoder features, which makes the training harder. Following the concatenation, a few 3×3 convolutions are applied to refine the features, finishing with another simple bilinear upsampling by a factor of 4.

2.6.2.2 Modified Aligned Xception

Inspired by the results of the Xception model [18] on ImageNet [43], and the modifications performed to the Xception module by the Microsoft Research Asia (MSRA) [1] team [20] (that further improved the module's performance in object detection tasks), the authors, decided to adapt the Xception model for semantic image segmentation. To do so, the authors made additional changes on the MSRAs modifications, namely:

1. A deeper Xception like the one in [20], but without modifying the entry flow network structure, for fast computation and memory efficiency;
2. All max pooling operations are replaced by depthwise separable convolution with striding. This allows for the use of atrous separable convolution to extract feature maps at an arbitrary resolution;
3. Extra batch normalization and ReLU activation are added after each 3×3 depthwise convolution (similar to MobileNet [29]).

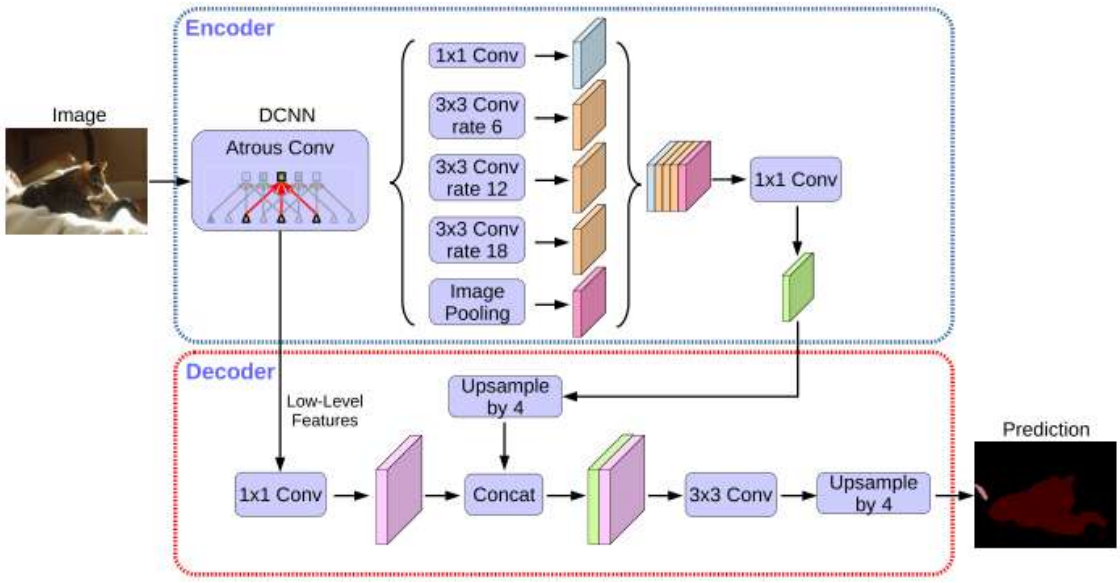


Figure 2.8: DeepLabV3+ encoder-decoder structure [17].

2.6.3 HRNet

Most of the recently developed classification networks, like Alex Net [35] and ResNet [25], gradually reduce the spatial size of the feature maps, connect the convolutions from high-resolution to low-resolution in series, and lead to a low-resolution representation, which is further processed for classification. However, for position-sensitive tasks (e.g., semantic segmentation, object detection, and human pose estimation) high-resolution representations are needed [51]. Other state-of-the-art methods, use the high-resolution recovery

[1] https://microsoft.com/research/lab/microsoft-research-asia/

process to raise the representation resolution from the low-resolution representation outputted by a classification or classification-like network. In addition, dilated convolutions are used to remove some down-sample layers and thus yield medium-resolution representations [14]. Wang et al. [51] propose a novel architecture: High-Resolution Net (HRNet), which is able to maintain high-resolution representations through the whole process. It starts from a high/resolution convolution stream and then gradually adds high-to-low resolution convolution streams. Lastly, the multi-resolution steams are connected in parallel. An example of a high-resolution network architecture can be seen in figure 2.9.

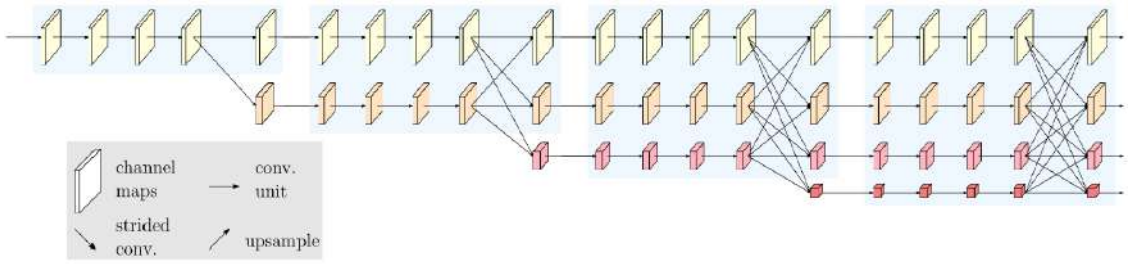


Figure 2.9: An example of a high-resolution network. Only the main body is illustrated, and the stem (two stride-2 3 × 3 convolutions) is not included. There are four stages. The 1st stage consists of high-resolution convolutions. The 2nd (3rd, 4th) stage repeats two-resolution (three-resolution, four-resolution) blocks [51].

The learned high-resolution representations are not only semantically strong but also spatially. This is due to two aspects of the network. First, the authors' approach connects high-to-low resolution convolution streams in parallel rather than series, which is more common. The way the proposed parallel multi-resolution convolutions work is as follows: The first stage is a high-resolution convolution stream. Then, gradually and one-by-one, high-to-low resolution streams are added, forming new stages. Lastly, the multi-resolution streams are connected in parallel. As a result, the resolutions for the parallel streams of a later stage consist of the previous stage's resolutions and an extra lower one. This parallel approach allows maintaining the high-resolution, rather than recovering high-resolution from low-resolution, making the representations spatially more precise. Second, the authors repeat multi-resolution fusions (the fusion module is illustrated in figure 2.10) to boost the high-resolution representations with the help of the low-resolution representations and vice versa. The goal of the fusion module is to exchange the information across multi-resolution representations. As a result, all the high-to-low resolution representations are semantically strong. This fusion scheme is different from most of the other schemes, which aggregate high-resolution low-level and high-level representations obtained by up-sampling low-resolution representations.

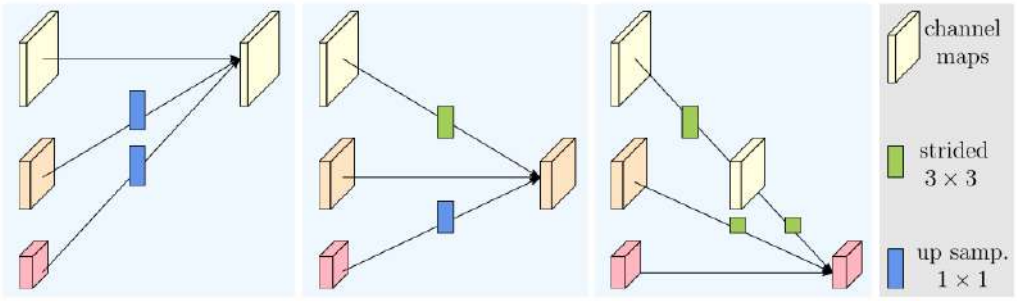


Figure 2.10: Representation of how the fusion module aggregates the information for high, medium and low-resolutions from left to right, respectively [51].

Simply put, the network connects high-to-low convolution streams in parallel and maintains high-resolution representations through the whole process, and generates reliable high-resolution representations with strong position sensitivity through repeatedly fusing the representations from multi-resolution streams.

2.6.4 OCR

Yuan et al. [57] aimed to augment the representation of one pixel by exploiting the representation of the object region of the corresponding class. A study done by the authors shows that such a representation augmentation scheme, when the ground-truth object region is given, dramatically improves the segmentation quality.

The authors' approach consists of three main steps. In the first one, the contextual pixels are divided into a set of soft object regions corresponding to a class. This division is done by supervised segmentation. Second, the representation of each object region is estimated by aggregating the representations of the pixels in the corresponding object region. Third, the representation of each pixel is augmented with Object Contextual Representation (OCR) [2] .

Note that the proposed OCR approach differs from the conventional multi-scale context schemes. First, the authors' approach to OCR differentiates the same-object-class contextual pixels from the different-object-class contextual pixels. This is different from multi-scale schemes (like ASPP from [15]) that only differentiate the pixels with different spatial positions. Figure 2.11 visually represents the difference between OCR context and multi-scale context. Second, their approach also structures the contextual pixels into object regions and exploits the relations between pixels and object regions. This is opposed to previous relational context schemes, that consider the contextual pixels separately and only exploit the relations between pixels and contextual pixels.

[2] OCR can be defined as the weighted aggregation of all the object region representations, with the weights calculated according to the relations between pixels and object regions.

(a) ASPP (b) OCR

Figure 2.11: Visual representation of multi-scale context (with ASPP) and OCR context. The context is for the pixel marked as red. (a) ASPP: The context is a set of sparsely sampled pixels marked with yellow and blue. The pixels with different colors correspond to different dilation rates. (b) OCR: The context is expected to be a set of pixels lying in the object (marked with color blue) [57].

2.7 Spectral Image Classification

HSI classification using CNNs has seen an increased adoption in current literature. Because of the richness of spectral information, it has a wide range of applications in various fields, such as land-cover detection, agricultural development, and environmental protection (such as detection of IAS). Approaches to the problem range from the use of SVMs and RFs to 3D-2D CNNs and Recurrent Neural Networks (RNNs). This section presents some of the state-of-the-art models regarding HSI classification.

2.7.1 SSDGL

Many of the novel networks applied to HSI classification can achieve great results when provided with sufficient labeled data. However, these methods only consider the labeled samples and ignore the spectral-spatial information of unlabeled samples. This can pose a problem since the available hyperspectral data is imbalanced most of the time. This is due to how difficult it can be to identify land-covers by visual interpretation. Thus, Zhu et al. [60] propose the Spectral-spatial Dependent Global Learning (SSDGL) framework to extract the deep spectral-spatial features and solve the sample problem of insufficiency and imbalance. The proposed method is an ensemble learning method that combines spectral, structural, and semantic features. The most discriminative feature representations are learned by the Global Convolutional Long Short-term Memory (GCL) integrated with the Global Joint Attention Mechanism (GJAM).
The SSDGL framework solves the insufficient and imbalanced data problem by using a hierarchically balanced sampling strategy that is utilized to generate stochastic hierarchical training sample data. This sampling strategy reduces the overall training times and speeds up model convergence. The weighted softmax with cross-entropy loss is introduced to reduce the weight of easy-to-classify samples so that the model focuses more on

hard-to-classify during training.
To extract the detailed spectral-spatial information of the whole image, GCL is proposed to capture the long short-term spectral dependent features and leverage the convolutional kernel to extract interrelations among the local pixels. The GCL module can effectively distinguish similar land covers by extracting the intrinsic spectral-spatial dependency.
Lastly, a GJAM is used to extract the most discriminative feature representation further. This module is composed of a spectral attention mechanism and a spatial attention mechanism. The spectral attention mechanism can selectively emphasize informative spectral features and suppress less-useful ones. The spatial attention mechanism is introduced to extract the short-term spatially dependent features and emphasize the key regions.

2.7.2 SpectralNet

SpectralNet [12] is a HSI classification model proposed by Chakraborty et al. [12], that aims to overcome some of the limitations and constraints of previously proposed models. Other proposed approaches besides 3D-2D CNNs and FuSENet [41] do not consider the spectral and spatial features together for HSI classification, thereby resulting in poor performances. However, 3D CNNs are computationally heavy, and 2D CNNs do not consider multi-resolution processing of images. SpectralNet is a wavelet CNN based on the 2D CNN for multi-resolution HSI classification. Computing a wavelet transform to extract the spectral features is less computationally demanding than a 3D CNN. The extracted spectral features are then fed to the 2D CNN for spatial feature extraction, resulting in a spacial-spectral feature vector. The authors have proved this to be a better approach for multi-resolution HSI data.
A simplified version of how the model works is the following (note that this model takes only one HSI as input. This is a limitation, not allowing for multi-image classification): the input HSI with dimensions $M \times N \times R$ is sent through a FA layer to reduce the dimension to $M \times N \times B$. This reduces the training time by up to 60%. The model uses FA instead of the more common PCA because the first is able to describe the variability among the different correlated and overlapping spectrum bands, which helps make the model classify similar examples better. After the FA step, overlapping 3D patches of size $S \times S \times B, S < M$ are extracted from the pre-processed HSI and sent to SpectralNet. The patches are then decomposed by a four-level wavelet transform into sub-bands. These sub-bands are then sent through a convolution layer to learn the spectral and spatial features. Since that SpectralNet is a multi-resolution CNN, the convolution is performed by a pair of channels (low and high).

2.8 Conclusion

This chapter presented an overview of remote sensing, IAS in aquatic ecosystems, map accuracy assessment, and some methods used in computer vision. These are fundamental 'preliminary' concepts that support our thesis work. We started by introducing the concept of RS and how it can be used to map IAS and covered the general process of detecting and mapping IAS using RS. Furthermore, we discussed how IAS can affect aquatic ecosystems and the many techniques that can be used to identify and differentiate targeted species. Then we presented some of the most common pre-processing operations in RS and metrics to assess the accuracy of the produced maps.
Lastly, we presented classical algorithms like RFs and SVMs, followed by state-of-the-art deep learning models used in semantic segmentation tasks. The goal of this chapter was to understand the fundamentals of RS and the many techniques that can be used to identify and differentiate IAS. This will allow us to carry out a more rigorous work during the stages of creating the dataset and evaluating the results of the proposed model.

Chapter 3

Related Work

In this chapter, we review and discuss both the used techniques and the achieved results by other authors. Section 3.1 presents the work done by Bolch et al. [9] with the assessment of the performance and feasibility of using drone-mounted sensors to detect invasive aquatic vegetation. Section 3.2 analyses the work done by Khanna et al. [31] studying the changes in water primrose invasion in an estuarine ecosystem.

3.1 Performance and Feasibility of Drone-Mounted Imaging Spectroscopy for Invasive Aquatic Vegetation Detection

Bolch et al. [9] identified water hyacinth (Eichhornia crassipes) and water primrose (Ludwigia spp.) with better results than the ones achieved by Airborne Hyperspectral Imaging (HyMap), using only Drone-Mounted Imaging Spectroscopy. This will also be the approach we will take in this thesis.

3.1.1 Study Site

The affected study site was the Sacramento-San Joaquin River Delta, the upstream of the San Francisco Estuary, which is the largest tidal freshwater estuary in the western United States. This body of water is of most importance since it supports agriculture and a lake-like wetland, which is the habitat for numerous species. It is also one of the most invaded ecosystems globally, threatening water quality, commerce, recreation, and even native species.
The Center for Southeastern Tropical Advanced Remote Sensing (CSTARS) [3] tasked a MAV-mounted imaging spectrometer to collect images in April 2019 across the whole Delta. The authors used this opportunity to collect concurrent UAV-mounted imaging spectroscopy to compare its mapping capabilities. The UAV study area consisted of two areas with roughly 200×200 m that contained two invasive aquatic macrophytes, as well as other common species found throughout the Delta. Two major floating invasive species of concern in the Delta are water hyacinth (Eichhornia crassipes) and water primrose (Ludwigia spp.). Ludwigia spp. is very similar to Ludwigia peploides, thus we can consider them as being the same for the purpose of our thesis. Its amphibious capability, and fast growth rates have made it a threat to the Delta, endangering native species and wetland

[3] https://rsmas.miami.edu/research/centers/cstars

restoration projects in the region. It is also poses a threat to humans because primrose mats provide habitats for mosquitoes transmitting the West Nile virus. Table 3.1 shows the classes of interest and descriptions for the UAV.

Table 3.1: Classes of interest and descriptions for the UAV and manned flights [9].

Map Class	Description
Unclassified	Unclassified land cover area outside of analysis.
Bare Ground	Asphalts, gravel, levee riprap, and bare soil.
Emergent Vegetation	Cat tail (Typha spp.), common reed (Phragmites australis), giant reed (Arundo donax), and tule (Schoenoplectus spp.).
Water Hyacinth	Water Hyacinth (Eichhornia crassipes).
Water Primrose	Water Primrose (Ludwigia spp.).
Riparian	Shrubs and trees in the area including willow species (Salix spp.).
Submerged Aquatic Vegetation	Numerous species; dominant ones include: Brazillian waterweed (Egeria densa), coontail (Ceratophyllum demersum), and watermilfoil (Myriophyllum spicatum).
Water	Water.
Other Vegetation	Species or cover not observed in the UAV study region including pennywort (Hydrocotyle spp.), and mosquito fern (Azolla spp.).
Non-photosynthetic vegetation	Senescent or dead vegetation.

3.1.2 Data Collection and Equipment

The used sensor was the Headwall Nano-Hyperspec, mounted in a Da-Jiang Innovations (DJI)-M600Pro [4] UAV with a DJI Ronin gimbal [5] . The DJI-M600Pro is a six-rotor UAV system weighing 10 kg with a 1.133 m diagonal wheelbase. It has a maximum take-off weight of 15 kg and an approximately 16 min flight time at that weight. The Nano records radiance in 270 visible and near-infrared light bands across 400 to 1000 nm with 2.2 nm of spectral resolution. The flying altitude was set to 115 m, which results in a spatial resolution of 0.051 m (this result can vary slightly due to ground topography and wind conditions). The imagery was collected between 12:30 and 13:15 PDT on April 9^{th} 2019. Solar radiation was approximately 907 w/sq.mn and wind speed was roughly 11.4 kph during the flights. The UAV was flown over two regions, and sixteen flight lines were collected, eight flying into and eight flying away from the solar plane. The raw image cubes collected by the Nano were converted to radiance using a dark calibration of the sensor conducted preflight, then the imagery was orthorectified and converted to reflectance cubes using Headwall's Spectral View software.

[4] https://dji.com/matrice600-pro
[5] https://dji.com/ronin-mx

The HyMap sensor, operated by HyVista Corporation (Sydney, Australia) [6] , is a whiskbroom sensor system consisting of a silicon detector array and three Indium Antimonide (InSb) array modules that provide contiguous spectral sampling across the visible, near-infrared, and shortwave infrared regions. It is mounted on a gyro-stabilized platform, and the detector array has 512 pixels. From 9^{th} to 12^{th} of April 2019, HyVista flew HyMap over the Sacramento- San Joaquin Delta. The data were collected with a ground resolution of 1.7 m with a 20% overlap in flight lines. HyVista performed geocorrection and atmospheric correction using proprietary HyCorr software. Table 3.2 contains more information about both the used sensor and platform and HyMap.

Table 3.2: Sensor and Platform Specifications [9].

	HyMap	**Nano-Hyperspec**
Type	whiskbroom	pushbroom
Spectral Range	450-2480 nm	400-1000 nm
Number of bands	128	270
Spectral Resolution	15-18 nm	2.2 nm
Signal to Noise	$>$ 500:1	$>$ 15:1 (1000 nm) $<$ 140:1 (550 nm)
Spectral Resolution [7]	1.7 m	0.051 m
Swath Width (FOV °)	61.3	15.85
Operational Altitude	$>$ 458 m [8]	$<$ 122 m [9]
Platform	1975 Rockwell International 500-S	DJI-M600P

3.1.3 Classifier

The chosen classifier was a RF, as these models are a widespread choice for RS. SVMs are another popular classification method that has successfully been used to map species. The RF models were constructed using the *caret* and *randomForest* packages in the *R* programming language. Despite often having better classification accuracies than other models, they lack direct quantification error, which is essential to quantify RF uncertainty. A bootstrapping procedure of building multiple random forests for each model was used to account for this. It allows capturing the range of accuracies of the RF, where the model ran randomly and selected a different sample of training and independent test data. After bootstrapping the RF models, the accuracy metrics of each model were examined, including overall accuracy, producer's accuracy, user's accuracy, and well as Cohen's Kappa statistic.

[6] https://hyvista.com

[7] Spatial resolution is not only dependent on the sensor but also the flight altitude.

[8] Typical lowest safe altitude according to Federal Aviation Administration (FAA).

[9] Maximum altitude for UAV without FAA approval.

3.1.4 Results

The map-making model selected for classification of the Nano imagery had an overall accuracy of 94.1%, performing better than the HyMap classification for 2019, with an overall accuracy of 85.7%. The Kappa Coefficient results were 92.6% for Nano and 83.0% for HyMap 2019. Tables 3.4 and 3.3 include the user's and producer's accuracies, respectively, for Nano and HyMap. Further results and discussions can be found in the article [9].

Table 3.3: Producer's accuracies [9].

Species	Nano Accuracy	HyMap Accuracy
Water hyacinth	87.5%	93.2%
Water primrose	100%	90.4% (high density) / 94.6% (low density)

Table 3.4: User's accuracies [9].

Species	Nano Accuracy	HyMap Accuracy
Water hyacinth	100%	89.9%
Water primrose	50%	94.9% (high density) / 94.4% (low density)

3.1.5 Conclusions

The work done by Bolch et al. [9], proves that it is possible to remotely detect and identify water primrose using data captured with UAV-mounted sensors. The authors used a simpler and cheaper setup than HyMap, and yet achieved better results. This work will be used as a baseline for our thesis, as we plan to use a similar approach in detecting Ludwigia peploides remotely, by using data captured with a drone-mounted sensor. Although, instead of using a SVM classifier we will use a neural-based semantic segmentation model.

3.2 Water Primrose Invasion Changes Successional Pathways in an Estuarine Ecosystem

Like the article [9] analyzed in section 3.1, the article by Khanna et al. [31] also focuses on studying the invasiveness of Water primrose (Ludwigia spp.) on the Sacramento-San Joaquin River Delta. We will be analyzing this article precisely because the study site and species is the same, thus eliminating any variables that could emerge from studying different sites or plants. This way, we can focus on analyzing the different methods of data collection, pre-processing and different models, and to some degree, guaranty that any difference in the results of both articles, come from the used methodologies.

3.2.1 Study Site

As the article [9], the study site was also the Sacramento-San Joaquin River Delta, the upstream of the San Francisco Estuary, which is the largest tidal freshwater estuary in the western United States. The focus area was composed of two sections of the Delta.

The first section is Liberty Island in the northwest Delta, a naturally restored freshwater tidal wetland of 21 km^2 created by flooding a reclaimed agricultural tract. The flooding has produced a shallow wetland with spatially variable tides and flows, and temporally variable seasonal and yearly fluctuations in water levels, depending on the upstream freshwater supplies.

The second area is the Central Delta, characterized by its tidally active dynamic marshes. This area is composed of meandering channels and inundated islands, all created by land reclamation and the construction of levees in the early 1900s. Inundated islands arise from levee failure over time. This has created a diverse system of channels and large expanses of water with varying bathymetry and water velocity. Over recent years, the Central Delta has experienced significant changes in its vegetation communities, with variable extents of invaded submerged plant communities and dynamic floating communities [33].

3.2.2 Data Collection and Equipment

Liberty Island and the Central Delta imagery was collected by both Airborne Visible and InfraRed Imaging Spectrometer — next generation (AVIRIS-ng) [10] and the HyMap sensor. In June of 2004 and 2008, spectroscopy data from the HyMap sensor (126 bands: 400 – 2500 nm, band width: 10 – 15 nm) were collected over the Delta at 3 m ground resolution by HyVista Corporation (Sydney, Australia). In Fall of 2014 and 2016, AVIRIS-ng data (430 bands: 350 – 2500 nm, band- width: 5 – 7 nm) were collected over the Delta at 2.5 m ground resolution by the Jet Propulsion Laboratory (Pasadena, California, USA) (JPL) [11] . Additional information about the image acquisition flights can be found in Table 3.5. Data were collected in 2 h windows before or after solar noon to minimize sunlight. Furthermore, close to low tide to minimize water column height over submerged vegetation.

Table 3.5: Additional information about the image acquisition flights [31].

Year	Dates	Sensor	No. Flightlines	Pixel size (m)
2004	6/25 to 7/9	HyMap	65	3
2008	6/29 to 7/7	HyMap	48	3
2014	11/14 to 11/25	AVIRIS-ng	60	2.5
2016	10/8 to 10/9	AVIRIS-ng	22	2.5

[10] https://avirisng.jpl.nasa.gov

[11] https://www.jpl.nasa.gov

Two side notes to consider: Although image acquisition occurred in two different seasons, the authors believe this is not problematic because water primrose shows active growth from June through October, and senescence occurs in November. Also, the analysis for this study included only the 22 common flight-lines present in all years.
Both HyMap and AVIRIS-ng data were atmospherically corrected to surface reflectance by HyVista and JPL, respectively. Preliminary geocorrection of the imagery was also completed by both corporations using on-board GPS and inertial navigation instruments obtained concurrently with the overflights. Images georeferenced based on this information often suffered from residual misalignment of $2 - 4$ pixels (personal observation by the authors [31]). The authors also performed a second level of geocorrection on the HyMap data using an orthorectification algorithm from Analytical Imaging and Geophysics.

3.2.3 Classifier

A RF was the chosen classifier by the authors. They had to use multiple techniques to capture reflectance properties across different regions of the electromagnetic spectrum and represent different biochemical properties of the plants. To capture plant water content and cellulose, they calculated band indices and continuum removals over water and cellulose absorption features centered at 980 nm, 1200 nm, and 2100 nm wavelength. To estimate the proportion of water, soil, non-photosynthetic vegetation, green vegetation, and submerged vegetation within a pixel, the authors used spectral mixture analysis [6]. They also created a spectral library of all emergent and floating species and used it to run a spectral angle mapper algorithm to detect species identity based on the angles between reflectance in consecutive bands and regions of the electromagnetic spectrum [28].
The above-mentioned inputs were used as input variables in a RF algorithm to classify: water, submerged, water primrose, water hyacinth, emergent, and non-photosynthetic vegetation (a term for the dry, non-green plant materials in the image).
The three floating species, water primrose, water hyacinth, and pennywort, were classified at the genus [12] level to focus on the impact of water primrose on other floating species and on the emergent and submerged plant communities. Regarding to the submerged species, it is tough to differentiate them, given that water absorbs almost all near-infrared and short-wave-infrared electromagnetic radiation [27, 44]. More specifically, the less abundant native species can be differentiated, while some non-native are more difficult to differentiate with airborne spectroscopy data. This results from their higher variability in the spectral signatures due to the broader range of environments they can survive and persist in [44]. Because of this, the authors decided not to differentiate between native and non-native submerged species, considering them as one single class. This should not pose a problem, as both the native and non-native submerged species respond similarly to light limitations imposed by the presence of water primrose.

[12] A genus is a taxonomic category ranking used in biological classification that is below family and above species. Species exhibiting similar characteristics comprise a genus.

3.2.4 Change Detection

The authors also calculated Change Detection (CD) statistics for the time steps 2004-2008, 2008-2014, and 2014-2016. The co-registration step between images is critical for detecting change across multiple years. It has been shown that a sub-pixel registration accuracy of one-fifth of a pixel can lead to CD errors as high as 10% [32]. The Optimal Scale Change Detection (OSCD) algorithm has a way to overcome this limitation. The algorithm is relatively robust to minor co-registration errors between images because it detects change at a coarser spatial scale than the spatial resolution of the imagery. Using this method in a previous study [32], the authors determined the optimum scale of CD as 30 m for the HyMap 2004–2008 data. To be consistent across years, they maintain this scale for all years. More details about the co-registration steps and change detection can be found in the article [31].

3.2.5 Results

Water primrose has increased fourfold in the two study areas of the Delta between 2004 and 2016, from 122 ha to 471 ha. The increase was slower from 2004 to 2014 (on average 12.7 ha per year), but it has accelerated between 2014 and 2016 (110.9 ha per year), and it was especially swift in Liberty Island (as can be seen in Figure 3.1 and Table 3.6).
The overall accuracy and Kappa coefficients of the RF classification for all four years (2004, 2008, 2014, and 2016) are shown in Table 3.7. Accuracies were over 85%, and Kappa coefficients were over 82%, indicating excellent agreement between field data and image classification and, therefore, a successful classification for change detection.

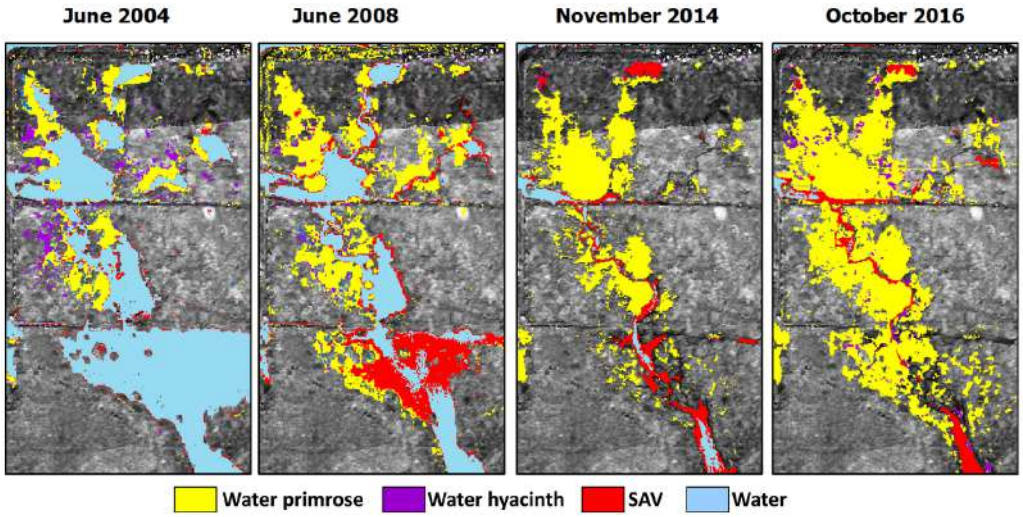


Figure 3.1: Water primrose expansion into open water and submerged vegetation habitat (June 2008 and November 2014) and finally into emergent marsh habitat (October 2016) [31].

Table 3.6: Water primrose cover in hectares in Central Delta and Liberty Island from 2004 to 2016 [31].

	Water primrose cover in hectares			
Location	**2004**	**2008**	**2014**	**2016**
Central Delta	84.8	106.5	216.2	388.3
Liberty Island	37.0	51.3	33.2	82.9
Total	121.8	157.8	249.4	471.3

Table 3.7: Kappa coefficients and overall accuracies for years of imagery classified [31].

Year	Overall accuracy (%)	Kappa coefficient (%)	Primrose kappa (%)
2004	86.9	84.0	82.0
2008	93.1	91.1	97.3
2014	86.7	83.5	89.3
2016	88.8	86.4	86.9

3.2.6 Conclusions

Unlike the work in [9] the goal of Khanna et al. [31] was not to prove the feasibility of using data captured with drone-mounted sensors, but rather to study the proliferation of the infestation. This is also one of the goals of our thesis, as we want not only to remotely detect Ludwigia peploides but also be able to assess the degree of the infestation at the Reservoir of the Toulica Dam.
Khanna et al. [31] used data collected from two sensors mounted to MAVs, one being the HyMap also used in [9]. The authors used a RF classifier, but instead of training the model with only 'plain' images, they also used multiple techniques to capture reflectance properties across different regions of the electromagnetic spectrum and represent different biochemical properties of the plant. Furthermore, they created a spectral library of all emergent and floating species. They used it to run a spectral angle mapper algorithm to detect species identity based on the angles between reflectance in consecutive bands and regions of the electromagnetic spectrum. Overall the authors obtained great classification results, and where able to study the evolution of the infestation.

3.3 Conclusion

The use of Artificial Intelligence (AI) in RS is a niche area, and thus the available related work is very limited when compared to other areas of AI. Most of the related work we found has mainly tree "problems". They are old and rely on techniques like manual feature extraction, which is outdated; they focus on another species than Ludwigia p.; they do not use aerial data. Nevertheless, the two articles we analyzed are great baselines for both proving the feasibility of using drone-mounted imaging and the study of IAS infestations using RS. We will try to improve on the work of the analyzed articles to accurately assess the degree of the infestation at the Reservoir of the Toulica Dam. Further chapters, will cover the used sensors, platforms and techniques.

Chapter 4

Data Sets

In this chapter, we present an overview of the existing public data sets with hyperspectral remote sensing scenes and our LudVision data set. Section 4.1, goes over the main characteristics and explores some advantages and limitations of the currently available data sets, in addition to some image examples, to illustrate each database. Section 4.2, analyzes the available platforms and sensors and their respective advantages and disadvantages. Section 4.3, presents the LudVision data set created due to the lack of a public data sets regarding Ludwigia peploides. It introduces the study site and the IAS Ludwigia peploides, which is the subject of this thesis. Section 4.3, also details the sensor and platform chosen for data collection, the collection process itself, and the annotation process.

4.1 Related Datasets

This section, covers some of the existing hyperspectral data sets. Despite not existing any publicly available data sets regarding Ludwigia peploides, or any similar aquatic species, analyzing existing hyperspectral data sets will help understand their general specifications.

4.1.1 Indian Pines

Indian Pines is a scene captured by Airborne Visible and InfraRed Imaging (AVIRIS) sensor [13] over the Indian Pines test site in North-west Indiana, USA. It consists of 145×145 pixels and 224 reflectance bands in the wavelength range $400 - 2500$ nm. The scene contains two-thirds agriculture and one-third forest or other natural perennial vegetation. There are two major dual-lane highways, a rail line, as well as some low-density housing, other built structures, and minor roads. Since the scene is taken in June, some of the crops present are in early stages of growth with less than 5% coverage. The ground truth has sixteen classes, detailed in table 4.1.

4.1.2 Salinas

The AVIRIS sensor also collected the Salinas scene over Salinas Valley, California, USA. It has a high spatial resolution of 3.7 meters per pixel, and the covered area comprises 512 lines by 217 samples. This scene includes vegetables, bare soils, and vineyard fields, in a total of sixteen ground-truth classes. More detail about the classes and samples can be viewed in table 4.2.

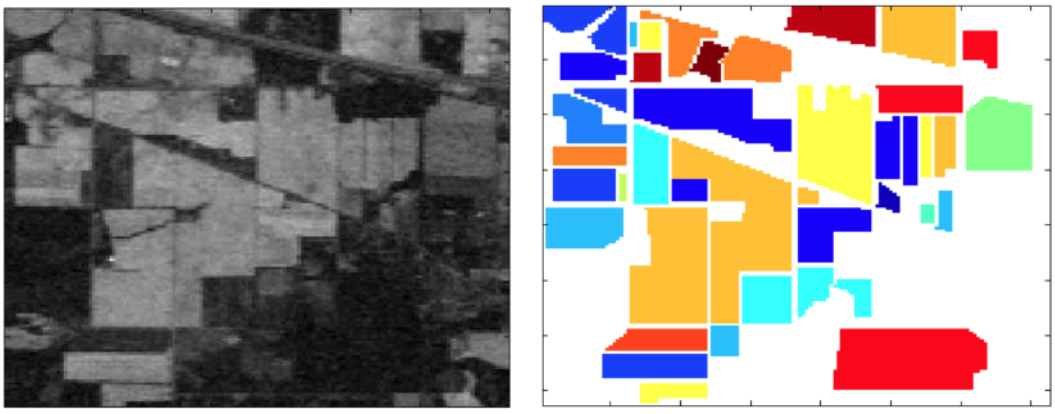
[13] https://aviris.jpl.nasa.gov

4.1.3 Pavia Centre and University

These are two scenes acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor over Pavia, in northern Italy. Pavia Centre is an image with 1096×1096 pixels with 102 bands, and Pavia University is an image with 610×610 pixels with 103 bands. Both have a spatial resolution of 1.3 meters per pixel and a ground truth of nine classes (additional detail about the Pavia Centre and Pavia University can be visualized in tables 4.3 and 4.4, respectively).

Table 4.1: Ground truth classes for the Indian Pines scene and their respective samples [2].

#	Class	Samples
1	Alfalfa	46
2	Corn notill	1428
3	Corn mintill	830
4	Corn	237
5	Grass pasture	483
6	Grass trees	730
7	Grass pasture mowed	28
8	Hay windrowed	478
9	Oats	20
10	Soybean notill	972
11	Soybean mintill	2455
12	Soybean clean	593
13	Wheat	205
14	Woods	1265
15	Buildings Grass Trees Drives	386
16	Stone Steel Towers	93

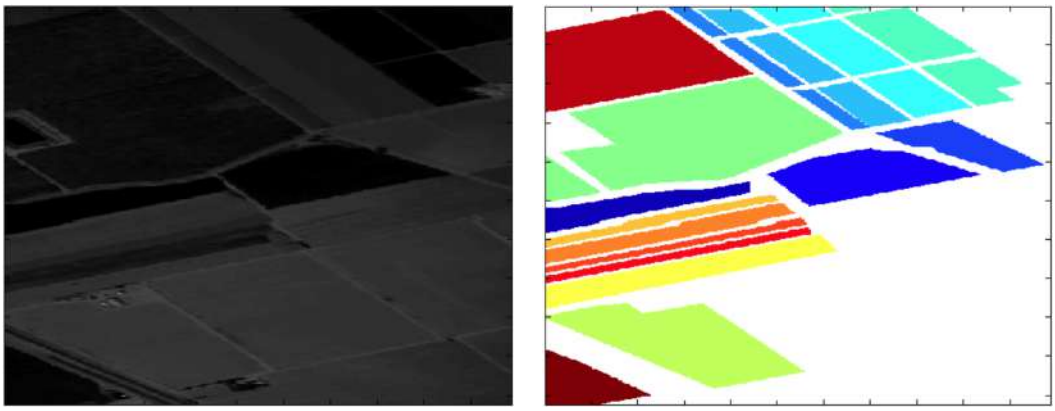


(a) (b)

Figure 4.1: Sample band (a) and ground truth (b) of Indian Pines data set [2].

Table 4.2: Ground truth classes for the Salinas scene and their respective samples [2].

#	Class	Samples
1	Brocoli green weeds 1	2009
2	Brocoli green weeds 2	3726
3	Fallow	1976
4	Fallow rough plow	1394
5	Fallow smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes untrained	11271
9	Soil vinyard develop	6203
10	Corn senesced green weeds	3278
11	Lettuce romaine 4wk	1068
12	Lettuce romaine 5wk	1927
13	Lettuce romaine 6wk	916
14	Lettuce romaine 7wk	1070
15	Vinyard untrained	7268
16	Vinyard vertical trellis	1807



(a) (b)

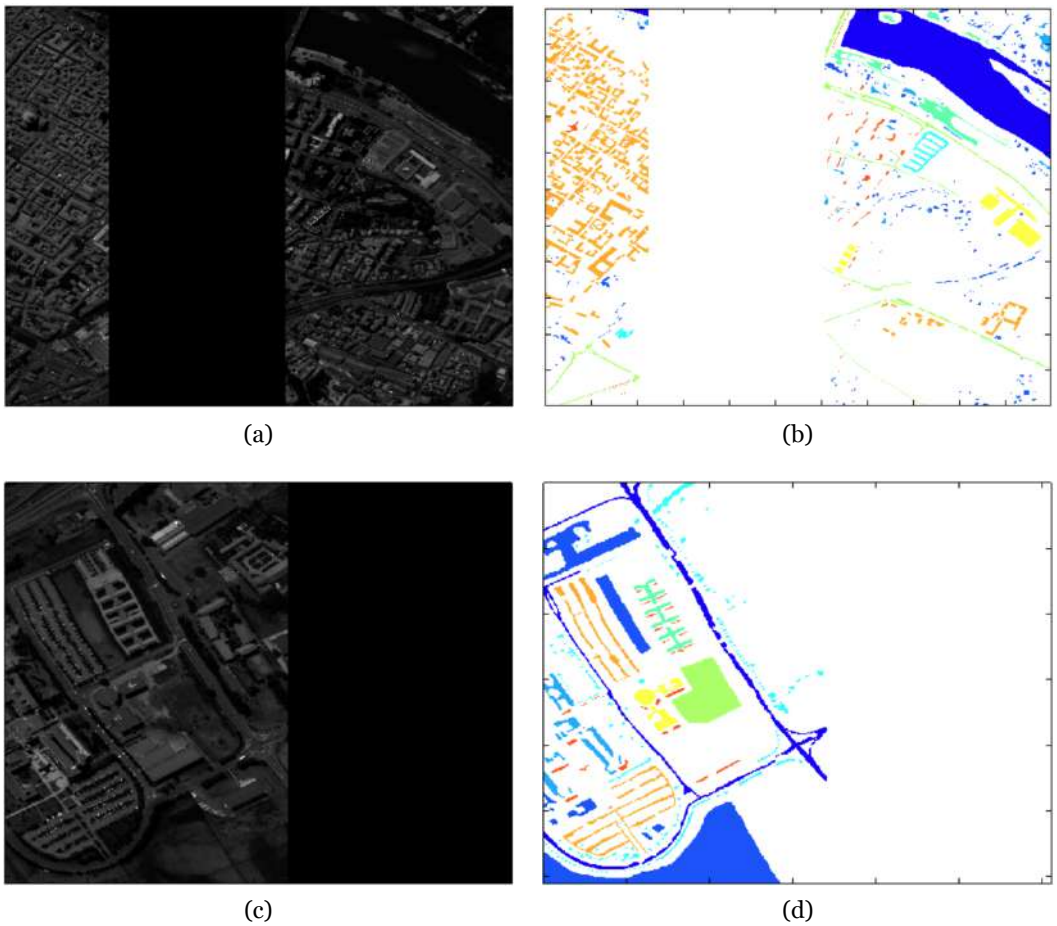
Figure 4.2: Sample band (a) and ground truth (b) of Salinas data set [2].

Table 4.3: Ground truth classes for the Pavia Centre scene and their respective samples [2].

#	Class	Samples
1	Water	824
2	Trees	820
3	Asphalt	816
4	Self-Blocking Bricks	808
5	Bitumen	808
6	Tiles	1260
7	Shadows	476
8	Meadows	824
9	Bare Soil	820

Table 4.4: Ground truth classes for the Pavia University scene and their respective samples [2].

#	Class	Samples
1	Asphalt	6631
2	Meadows	18649
3	Gravel	2099
4	Trees	3064
5	Painted metal sheets	1345
6	Bare Soil	5029
7	Bitumen	1330
8	Self-Blocking Bricks	3682
9	Shadows	947



(a) (b)

(c) (d)

Figure 4.3: Sample band (a) and ground truth (b) of Pavia Centre data set, and Sample band (c) and ground truth (d) of Pavia University data set [2].

4.2 Sensor and Platform Availability

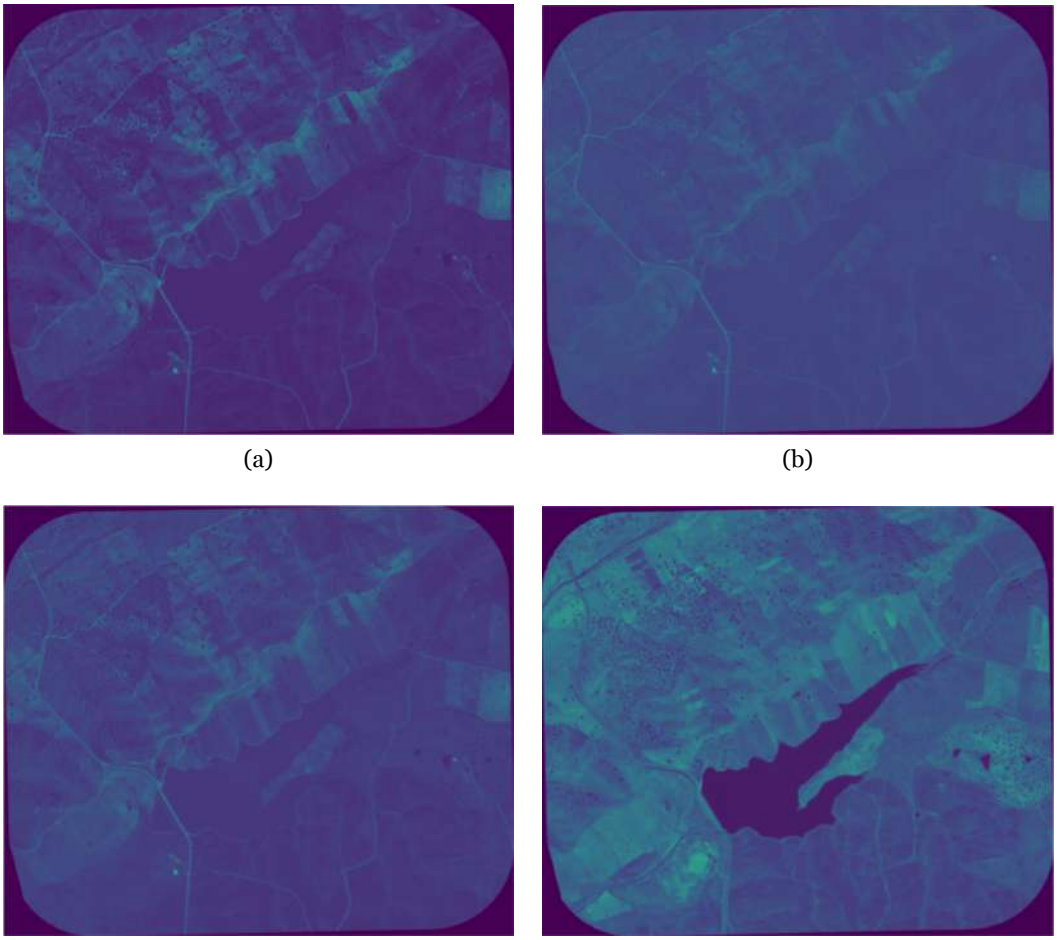
Many types of sensors and platforms can be used to collect multispectral and hyperspectral images [55]. With remotely sensed data, there are tradeoffs between spatial extent (size of the image), spatial resolution (pixel size), spectral resolution (number and range of visible and infra-red bands), and temporal resolution (frequency of data acquisition). Larger spatial extents allow for a broader mapping of IAS. However, spatial resolution tends to be low, allowing only to detect widespread infestations. Finer spatial resolution makes it more likely to detect individual species and early infestations. However, spatial extents and repeat temporal coverage are typically limited. Higher spectral resolution allows for differentiating plant pigments and chemistry in visible and infra-red bands. As a result, hyperspectral sensors are most commonly used for invasive plant detection. As we can see, no sensor can achieve high spatial, spectral, and temporal coverage over a broad spatial extent. Thus, a sensor and platform that suit the needs of each RS task have to be carefully chosen.
The following sections, analyze the available options and discuss their benefits and drawbacks. Lastly, they present our choice of sensor and platform.

4.2.1 Satellite

Satellites are well suited for large spatial extents but have a low spatial resolution, making them only feasible for widespread and abundant infestations. There are also have wait times for the satellite to fly above the targeted area for it to collect images. Depending on the satellite, the time gap between passes can be too long. This can pose significant drawbacks when trying to detect and control an IAS. If the passes are far apart, critical periods of the species life cycle can potentially missed, like, for example, spring season, where invasions tend to spread faster. Furthermore, acquiring satellite imagery is very expensive. The lack of spatial resolution, the time constraints due to satellite passes, and the high acquisition costs make the use of satellites unfeasible for our task. We acquired a satellite image covering our study site (4.5), to verify its usability for our task. The image has a resolution of 6065×5109 pixels and four bands (the individual bands can be seen in figure 4.4). As stated previously, the image has a great spatial extent but a low spatial resolution (around 3 m per pixel). Given that the infestation in the Reservoir of the Toulica Dam is at its early stages, it is impossible to identify the Ludwigia peploides accurately. This further proves that satellite images are not suited for our problem.

4.2.2 MAV Mounted Sensors

Another option for RS image collection is manned flights. MAVs have smaller spatial extents than satellites but compensate with higher spatial resolution. This makes them perfect for large study sites when higher resolution is needed. Examples of the use of this type



(a) (b)

(c) (d)

Figure 4.4: Red band (a), green band (b), blue band (c), and near Infra-red band (d) of the acquired satellite image.

of platform where presented in chapter 3, with AVIRIS-ng and HyMap. The major drawbacks of MAVs are the extensive yet required preparation phase, including flight plans and clearances, and the high costs. Furthermore, given the nature of these platforms, they only operate locally (here, we define locally as being in the same country or close neighboring countries). From our research, we could not find any MAV-mounted sensor operating in Portugal, thus making impossible the use of manned flights for collecting our data.

4.2.3 UAV Mounted Sensors

UAV and sensor technology improvements allow us to fill a gap between field surveys and the use of satellite images and manned flights. Although satellite and manned flights offer images with broader geographic coverage, those same images lack spatial resolution and are usually very expensive, as discussed. On the other hand, despite not covering a wide area, UAVs can provide images with outstanding resolution. Plus, they are more convenient than MAVs and satellites. UAVs have high operational flexibility and low cost

compared to other alternatives. Being almost ”launched on-demand” means UAVs can be launched virtually everywhere and at short notice, making them perfect for frequent, smallfootprint acquisitions. The UAV-mounted sensors offer much higher spatial resolution than aircraft-mounted imaging spectrometers but have lower spectral quality and a smaller spatial footprint due to lens specifications and UAV flight restrictions.

4.2.4 Multispectral and Hyperspectral Sensors

After analyzing the different types of platforms, it is also crucial to understand the different types of sensors and their specifications. In layman's terms, the difference between multispectral and hyperspectral sensors is their number of bands. Multispectral sensors usually have between 3 and 20 different band measurements in each pixel of the images they produce. Hyperspectral sensors contain upwards of 100 contiguous spectral bands, with some more advanced sensors having more than 200 bands. The numerous narrow bands of hyperspectral sensors provide a continuous spectral measurement across the entire electromagnetic spectrum and therefore are more sensitive to subtle variations in reflected energy. Images produced from hyperspectral sensors contain much more data than images from multispectral sensors and have a more significant potential to detect differences among land and water features. However, all these added bands come with a matched price-tag and complexity of use. Hyperspectral sensors can cost upwards of 100000 € and need special software to be operated that can also cost thousands of dollars in annual subscription plans. Given the complexity of some sensors and their respective sensors, there may also be the need for small courses to be able to operate the sensor and process the images. Furthermore, many hyperspectral sensors are heavy and need to be mounted on specialized MAVs or UAVs, which further increases the setup's cost and complexity.

4.3 LudVision Data Set

This section introduces our study site and give an overview of the targeted species (Ludwigia peploides). It also presents the chosen sensor and platform, as well as the data collection, pre-processing and annotation processes.

4.3.1 Contextualization

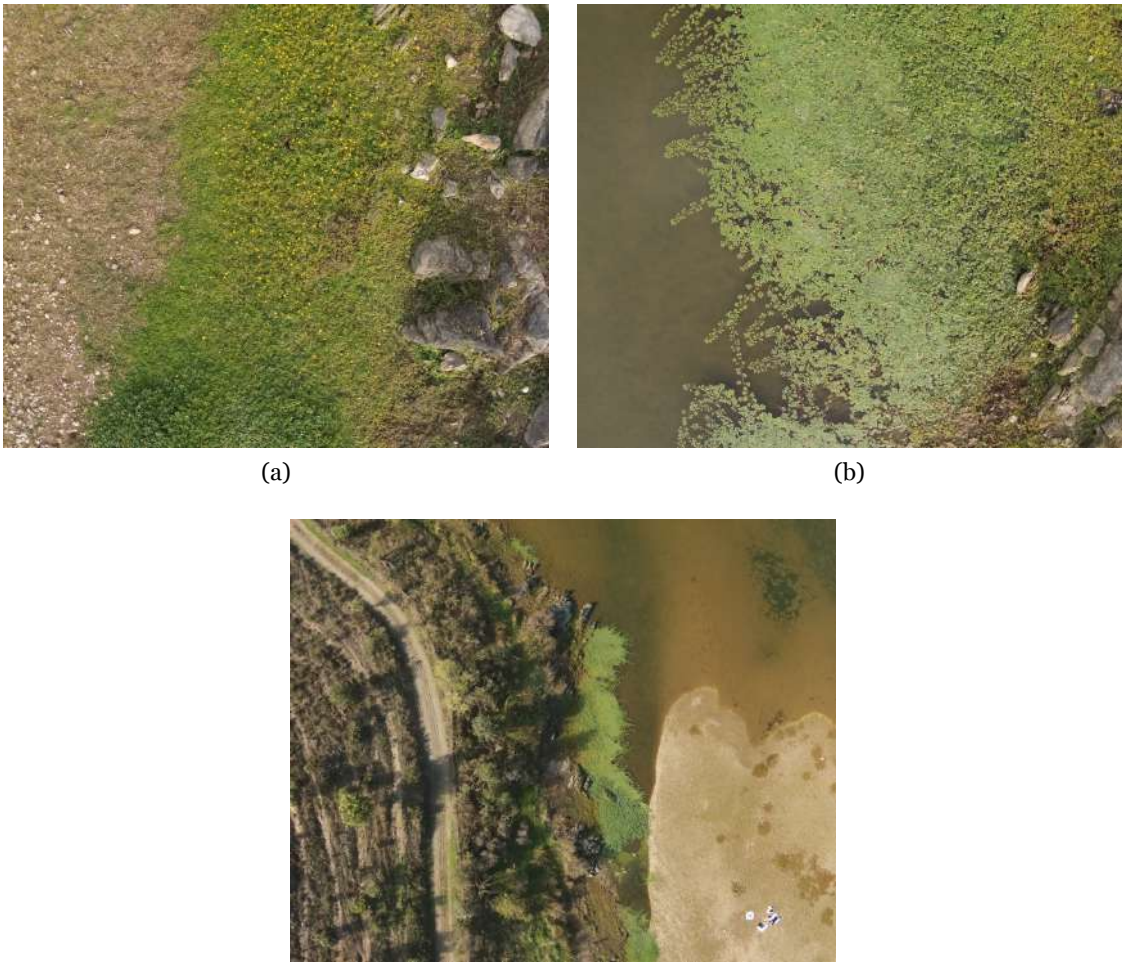
Due to the non-availability of public data sets containing the targeted IAS Ludwigia peploides, we created a data set from scratch. To build our data set, we first tried to use satellite imagery. However, due to some limitations in this type of data, which will be discuss later, we decided to use drone-mounted multispectral data. Our goal was to create a data set with good resolution and covers an extensive area. Using a drone to capture our data allowed for more flexibility and broader control of the various parameters like: height, camera angle, and overlap, beyond many others. We also had the support of a

biologist expert in aquatic species and ecosystems. With his help, we identified affected areas, made sure we were indeed capturing data of the suitable species, and understood some of the characteristics of the IAS. This allowed us to have a more comprehensive understanding of the species, which ultimately led to a data set tailored to our project's needs. Furthermore, the data set will be available for other researchers to use.

4.3.2 Study Site and Targeted Species

The study site is the Reservoir of the Toulica Dam (Zebreira, Portugal), located in the hydrographic basin of the Aravil river, a tributary of the Tagus. In 2020 the IAS Ludwigia peploides was spotted in the north-east part of the reservoir and has since spread, forming three big mantles. Recently it started spreading and forming small patches south-west of the initial infestation site. Ludwigia peploides is a species natural to South America that invades rivers, ponds, and rice fields. It can grow in deep waters, as a fully or partially submerged plant, and form floating mantles. When this happens, it prevents the entry of light affecting submerged species and blocking the water lines, affecting navigation, fishing, and recreational use. It competes for space by eliminating native species and producing substances that inhibit the germination and growth of other species. It reproduces vegetatively through stem fragmentation but also seeds. The species stems can grow between 10 cm and 3 m, hence its ability to form large mantles. The leaves are a bright green (that most of the time stands out from the rest of the present vegetation) and can have a lanceolate or oval shape. They measure between 2.5 and 3.8 centimeters, and both the stem and the leaves have different trichomes distributed over the surface. Ludwigia peploides also have solitary flowers with yellow petals, which measure from one to 1.5 cm in length, and which develop from tassels emerging from the upper part of the axillary bud. The species blooming period occurs from mid-spring to early fall, and during this period, the plant is easily identifiable. This is also the period where the species grows the most.

Figure 4.5: Satellite image of the study site (Reservoir of the Toulica Dam).



(a) (b)

(c)

Figure 4.6: Images from the Ludwigia peploides collected at the Reservoir of the Toulica Dam at different altitudes. (a) 15 m, (b) 40 m, and (c) 70 m.

4.3.3 Chosen Sensor and Platform

After extensive research on the available platforms and sensors, we concluded that the ideal combination would be a hyperspectral sensor mounted on a drone. This would allow for high spatial resolution, a wide range of bands, and a high operational flexibility. Given that our study site is small and the infestation is still at its early stages, we do not need to cover large extents. However, we did some research on the cost of acquiring a hyperspectral sensor mounted to a drone, and the costs ranged from 50000 € to 200000 €. This greatly exceeds the allocated budget for this project. Thus, we needed to find an alternative solution with high spatial resolution, high operational flexibility, and ease to set-up and operate.

After studying the available options on the market, we decided to acquire the DJI P4 Multispectral [14] . It is a drone based on the DJI Phantom 4 [15] , but instead of having a traditional

[14]https://dji.com/pt/p4-multispectral

[15]https://dji.com/pt/phantom-4

RGB camera, the P4 has six sensors. One RGB for visible light, and five monochrome sensors for multispectral imaging. Table 4.5 contains the drone, camera, and gimbal specifications.
It is sold as a ready-to-use package and is operated using the included remote, paired with any smartphone or tablet running the *DJI GO 4* app. This makes it very easy and intuitive to use. The app allows to define 'missions' that can be stored, and the ± 0.1 centimeter precise on-board Global Positioning System (GPS) guarantees that the images are collected at the same place and angle in future visits. This is great to ensure the consistency of our data. When defining a 'mission', one can define several parameters, the most relevant being: the area to be captured, the vertical and horizontal overlap ratio between images, altitude, drone speed, angle of the trajectory, and shutter mode (the drone can take images in scan mode and hover mode). In scan mode, the drone is constantly moving while taking pictures. In hover mode, the drone hovers while taking the pictures, allowing for a more steady image without motion-blur.

Table 4.5: DJI P4 Multispectral specifications.

Aircraft	
Takeoff Weight	1487 g
Max Ascent Speed	6 m/s (automatic flight); 5 m/s (manual control)
Max Descent Speed	3 m/s
Max Speed	31 mph (50 kph) (P-mode); 36 mph (58 kph) (A-mode)
Max Flight Time	Approx. 27 minutes
Operating Temperature	0° to 40° C (32° to 104° F)
Hover Accuracy Range	RTK enabled and functioning properly: Vertical: ± 0.1 m; Horizontal: ± 0.1 m RTK disabled: Vertical: ± 0.1 m (with vision positioning); ± 0.5 m (with GNSS positioning) Horizontal: ± 0.3 m (with vision positioning); ± 1.5 m (with GNSS positioning)
Gimbal	
Controllable Range	Tilt: -90° to +30°
Camera	
Sensors	Six 1/2.9" CMOS, including one RGB sensor for visible light imaging and five monochrome sensors for multispectral imaging. Each Sensor: Effective pixels 2.08 MP (2.12 MP in total)
Filters	Blue (B): 450 nm ± 16 nm; Green (G): 560 nm ± 16 nm; Red (R): 650 nm ± 16 nm; Red edge (RE): 730 nm ± 16 nm; Near-infrared (NIR): 840 nm ± 26 nm
Lenses	FOV (Field of View): 62.7° Focal Length: 5.74 mm (35 mm format equivalent: 40 mm), autofocus set at ∞ Aperture: f/2.2
RGB Sensor ISO Range	200 - 800
Monochrome Sensor Gain	1 - 8x
Electronic Global Shutter	1/100 - 1/20000 s (visible light imaging); 1/100 - 1/10000 s (multispectral imaging)
Max Image Size	1600×1300 (4:3.25)
Photo Format	JPEG (visible light imaging) + TIFF (multispectral imaging)

4.3.4 Data Collection

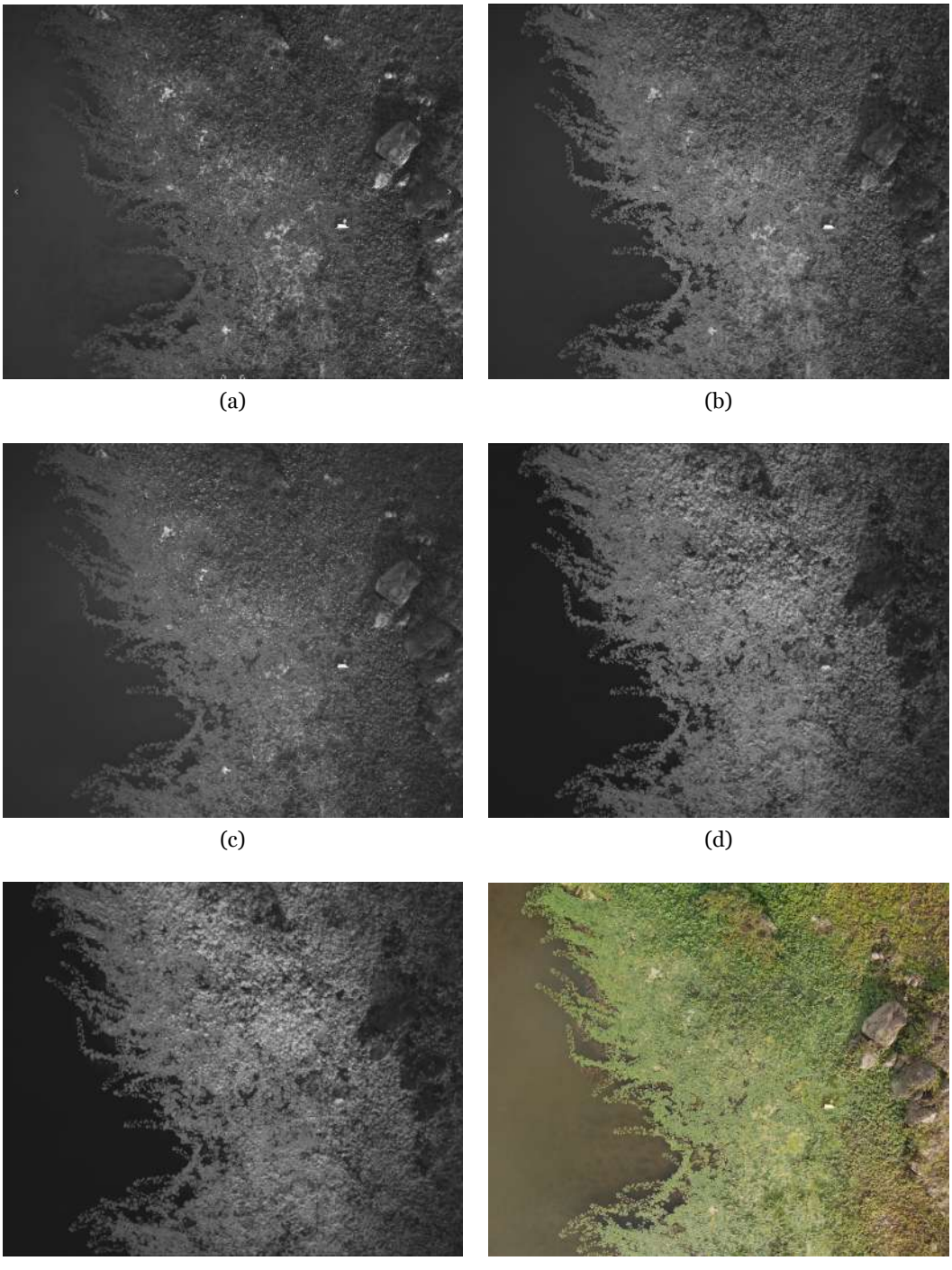
To collect our data, we visited the study site twice (October 11^{th} and 20^{th}, 2021). We captured all our data in hover mode because the drone is more stable, allowing us to point the camera straight down. This way, we can emulate the appearance of satellite images, which can be helpful in the later stages of our project. For example, we can create a mosaic of the entire study site, resulting in an image similar to satellite images with higher resolutions.
The data was collected at different altitudes, ranging from 10 m to 70 m. The goal with collecting data as different altitudes is that our model will learn features from a wide range of spatial resolutions. Thus, the final model should be more resilient to altitude variations, and the final goal is to be able to train our model with low altitude data and validate it with high altitude data (e.g., satellite data). The data was also taken at various times of the day to ensure that the model is resilient to variations in solar reflection. Note that no data was collected at solar noon, as it would result in overexposed images, due to the light being reflected from the lakes' surface. Table 4.6, contains more detailed information about the collected data.

Table 4.6: Altitude, time and number of images collected.

Altitude	Time	Number of images
10 m	11h - 12:45h 15:30h - 17h	435
15 m	11h - 12h	365
40 m	10h - 12:45h	135
70 m	11h - 12h	27

Every time the drone collects data, it is effectively taking six images, one RGB image in *.jpg* format and five monochromatic images in *.tif* format, one for each band, as can be seen in figure 4.7. The RGB image should only be used as a reference as it is easier to visualize the scene. The remaining five images are monochromatic and used to train the models. Tagged Image File Format (TIFF) is a lossless raster format hailed for its extremely high image quality. Often the format used by professionals in creative industries, these files require a large amount of storage space. TIFF is best for any raster images intended to be edited and is relied on to preserve quality. It offers options to use tags, layers, and transparency and is compatible with photo manipulation programs.
The TIFF images generated by our drone also include a header containing crucial information about flight details (altitude, coordinates), gimbal and camera information (like angle, ISO, exposure), and sun sensor information. It also includes some additional miscellaneous information, like the day the image was captured.
The included 'sun light sensor' is a feature of the DJI P4 Multispectral, that allows the drone to automatically correct the images according to the solar exposure. Usually, this step needs to be done in the pre-processing phase with specialized software, and before each flight, the sensor would need to be calibrated with a special calibration panel.

Overall, the collection step was straightforward, as all the calibration and optimal flight path are automatically calculated by the drone. We only needed to define the area, altitude, shutter mode, angle, and overlap ratios.



(a) (b) (c) (d) (e) (f)

Figure 4.7: Example of a red band (a), green band (b), blue band (c), red edge band (d), near infra-red band (e), and RGB image (f) collected by our drone.

4.3.5 Data Pre-processing

As stated in section 4.3.4, the drone already does a lot of the pre-processing on-board, like image reflectance calibration. Thus, the only pre-possessing left is to merge the individual bands into a single image. This way, we end up with a five-band image that can be fed to the model for training.
Merging the individual bands is not as straightforward as simply stacking the five bands and exporting them as a final image. The sensors are positioned in a 3×2 array, meaning they all have a unique perspective. Furthermore, we have to account for lens imperfections and distortions due to manufacturing tolerances. This way, if we just stacked the bands, the final image would be distorted. As can be seen in figure A.1, which contains an example of bands stacked before alignment, the image is distorted and looks overexposed. Figure A.2 contains an example of bands stacked after alignment, and as can be seen, the image is perfectly aligned without distortions.

The alignment was performed using the homography tools from *OpenCV* in *Python*. All the alignments are made relative to the blue band. The way it works is as follows:

1. Detect ORB features and compute descriptors using $cv2.ORB\_create$;
2. Match features between the two images using $cv2.DescriptorMatcher\_create$;
3. Sort matches by score;
4. Keep only the 5% best matches;
5. Draw top matches (for visual aid only, an example of the generated matches image can be seen in figure 4.8);
6. Extract location of good matches;
7. Find homography using $cv2.findHomography$;
8. Save array containing the homography data to a file.

This step is performed for blue-green, blue-red, blue-red edge, and blue-near infra-red. The homography data is saved in files to align all other images without having to recalculate them. Then once we have the homography data for each band relative to the blue band, we run another script that uses the files information to warp the perspective of the bands relative to the blue band using $cv2.warpPerspective$. Once all bands are aligned, they are stacked, and the final image is trimmed to 1400×1100 pixels and exported as a TIFF file. This step is done so that the final images do not have a 'frame' originated from shifting the images. One example of an image created before and after the alignment of bands, can be seen in the appendix A.1.
Note that we also generate a copy of the generated TIFF images in the *.png* format. We generate this copy because the annotation programs we use do not support the TIFF format. We will give more details about the annotation process in section 4.3.6.

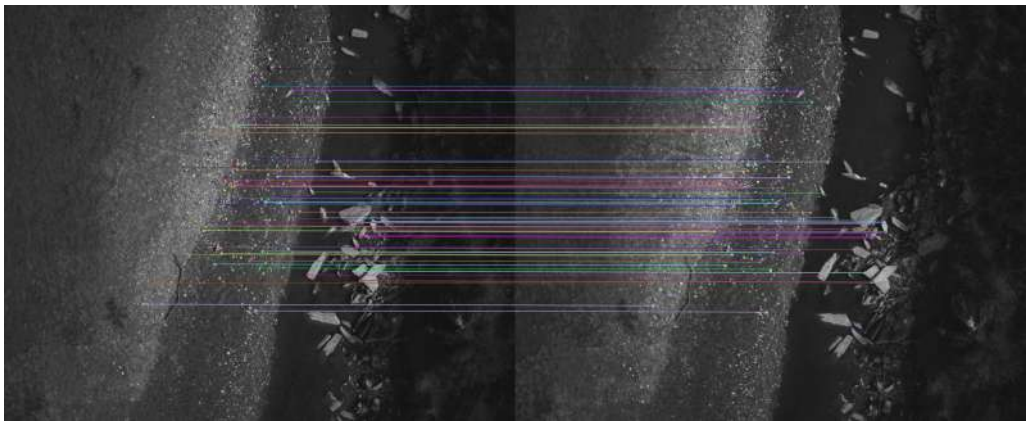


Figure 4.8: Example of the matches between the blue and green bands.

4.3.6 Annotation Process

For the annotation process, we use the *Hasty.ai* [16] tool. *Hasty.ai* is an excellent tool for annotating data with organic and complex shapes, which is precisely our case. It has Artificial Intelligence enabled tools that allow for assisted annotation, speeding up the entire process. It works by training a semantic segmentation model with images that are already annotated. Then it suggests annotations which can be edited, and the more images are annotated, the more accurate the model gets. After only about 60 images, the model annotated the remaining data with only minor errors, which were easily corrected by hand (note that despite being easy the editing of the suggested annotation takes some time). This assisting annotation tool is available after the first ten images are annotated and set to 'Done'. This tool significantly reduced the time dedicated to the annotation task. The only drawback of this tool is that it only exports data in the *COCO data set* and *Pascal VOC* formats. However, it allows the extraction of the semantic segmentation maps as a *.png* mask. We then used a custom *python* script to convert the data to the *Cityscapes* format.

As mentioned in section 4.3.5, when generating the images, we export them in both the TIFF and PNG formats. This is because the annotation tool do not support the TIFF format. So the labeling is done using the PNG images. However, the images are essentially the same, only in a different format, thus the pixels from the image in one format match the pixels in the image from the other format. So the labels for the image in the PNG format are also true for the images in the TIFF format and vice-versa. Because of this, we can use the masks created using the images in the PNG format to train the models that are given images in the TIFF format.

[16] https://app.hasty.ai

Figure 4.9: *Hasty.ai's* AI semantic assistant, training and validation chart.

Figure 4.10: Example of the annotation process.

Chapter 5

A New Method for Detection of Ludwigia Peploides in Multispectral Images

This chapter, presents the proposed method, the experimental results and the discussion of our experiments. As demonstrated in section 5.1, we started by testing the spectral radiance of Ludwigia peploides. This was to ensure that the used sensor captured enough spectral data to allow the isolation of the targeted species spectral signature. In section 5.2, we start by presenting the HRNet [51] model, and the justification for choosing this model to identify the presence of Ludwigia p. in the Toulica Dam. Then, we establish a baseline for our experiments, followed by the respective analysis and discussion of the results. After, we propose some modifications to the model and present the implementation details for the modifications. After implementing the modifications, we repeat the experiments. The results are analyzed and compared to the previous ones. Finally, we examine the model's performance in a qualitative way, by visually analyzing the output. This helped us to understand the scenarios in which our model is and is not able to identify the targeted species. This is an important step to address the model's issues in future work.

5.1 Testing for Spectral Radiance

After completing the pre-processing and annotation steps for our data, we assessed the spectral radiance. Our data was captured to allow us to leverage photophysiological measurements. Thus, we measured the radiance for Ludwigia p., rock, surrounding vegetation, and water. As expected, Ludwigia p. has a unique and distinct spectral signature, especially in the non-visible bands (Red Edge and Near Infra-red), as shown in figure 5.1. This supports our initial argument that a multispectral sensor would be enough to capture significant differences in the spectral radiance, allowing for the detection of Ludwigia peploides.

5.2 Tests and Results With HRNet+OCR

In previous chapters, we analyzed both preliminary concepts regarding RS and state-of-the-art semantic segmentation methods (chapter 2), and related work carried out by other authors (chapter 3). After completing our research, we concluded that we should take a ”hybrid” approach for our problem by combining both RS concepts and semantic segmentation. The reasoning behind our approach is that despite having a multi-spectral data set,

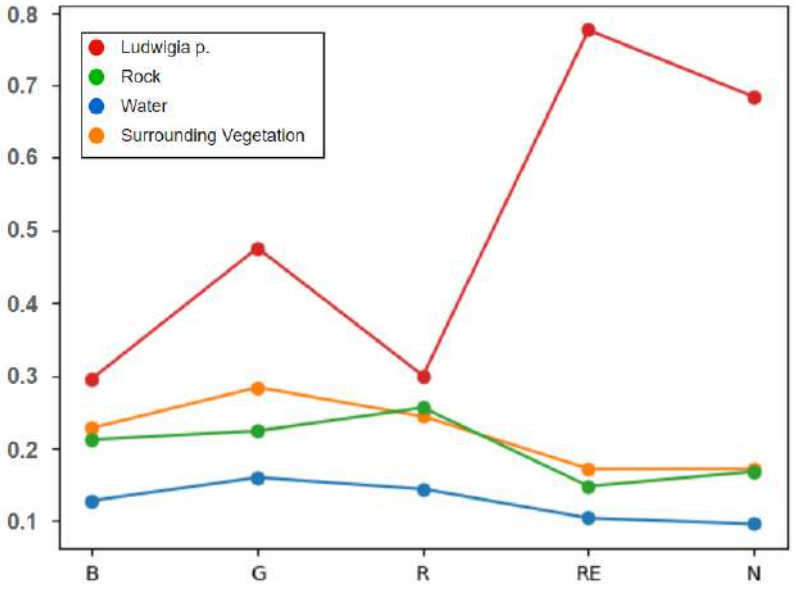


Figure 5.1: Reflectance values (in %) for each band corresponding to Ludwigia p., water, surrounding vegetation, and rock.

the number of bands in our data set is minute when compared to other studies. We only have five bands, whereas other authors had access to hyperspectral data, with considerably more bands. Thus, we believe that relying solely on the information of only five bands will not be enough to use models like the ones used by other authors [9, 31].

We propose the use of state-of-the-art semantic segmentation models that will be modified to take our data as input. Given that we only have two additional bands, compared to traditional RGB images we could not approach the problem as a standard remote sensing spectroscopy problem. By using semantic segmentation models, we can leverage both the capabilities of these models to recognize objects and the multi-spectral nature of our data. Fundamentally, the model will have the same behavior as usual but have access to the information five rather than bands, which should help it find meaningful features to detect the presence of Ludwigia peploides. Another point that made us consider using semantic segmentation models is that we want to identify the targeted species (Ludwigia peploides) as a whole, which means that the goal is to determine whether or not a given site is infected, and the degree of infestation. We have no interest in identifying individual plants. Plus, given the physiognomy of the species, that would be nearly impossible.

To test our theory and assess whether or not it is valid, we performed a series of tests. These tests will be our baseline results, which will then be compared to the results achieved after implementing the model's proposed modifications. After evaluating the model before and after the modifications, we will do a comparative analysis of the results.

The semantic segmentation model we chose was the HRNet [51] model in its HRNet + OCR [57] implementation. We chose this model for two reasons: First, the model can extract high-resolution representations, which are needed for position-sensitive tasks, like semantic segmentation. The high-resolution representations are attained by connecting

high-to-low resolution convolution streams in parallel rather than in series. Then, the representations are fused by a custom fusion module, which exchanges information across multi-resolution representations. This means that lower-resolution representations also contain information obtained from the high-resolution representations. This allows the model to extract better features for the images at lower resolutions. Given that images taken at high altitudes tend to have low resolution, it makes this model fitting for our data ; Second, the mentioned implementation integrates the OCR module [51]. This context module uses a set of pixels lying in the object instead of a set of surrounding sparsely sampled pixels. This is achieved by differentiating the same-object-class contextual pixels from the different-object-class contextual pixels and structuring the contextual pixels into object regions to exploit the relations between pixels and object regions. This allows setting the context for each pixel more accurately. Because Ludwigia p. typically forms large mantles at the water's surface, the module will have a larger area to extract the context for a given pixel. Our data set only has two labels (Ludwigia p. and background), making it easier to differentiate same-object-class contextual pixels from the different-object-class contextual.

As a side note, we did consider using existing spectral classification models analyzed in 2.7. Unfortunately, after doing some initial tests with them, we reached two conclusions, that led us to discard their use. First, they are poorly optimized for multi-image data sets. They were designed to deal with single image data sets (typically satellite gathered images). Second, the division of data in the train, test, and validation sets is poorly done, allowing the leakage of information between sets. This leakage will inevitably create misleading results when assessing the model's performance. The abovementioned issues could be fixed, but we do not believe they would have performed better than the HRNet + OCR model. Also, spectral classification models were designed to have hyperspectral data as input, which, as already mentioned, have more bands and, ultimately, more data per pixel than our data set.

5.2.1 Establishing a Baseline

To train the model on our data, a few modifications had to be made, especially on the input layer. Given that the model was originally designed to use traditional RGB imagery, we had to modify the input layer to be able to accommodate our data which has five bands (RGB plus a Near InfraRed and a RedEdge band).
We established a baseline by performing a series of experiments. We trained and tested the model on our data in different configurations. These baseline results will be compared to those achieved after implementing the proposed modifications to the model. In order to make it easier for the reader to understand the flow of the experiments, we created a flow chart (figure 5.2). The goal of the experiments is to assess the model's capability to identify Ludwigia p. from images captured at different altitudes. Also, we wanted to understand whether or not the model would benefit from progressive training. We define

progressive training as training the model using lower altitude images and progressively exposing the model to images captured at higher altitudes while also keeping some images from previous heights.
As stated previously in 5.2, the information from high-resolution representations is shared with low-resolution representations. This is due to the high-to-low resolution convolution streams, multi-resolution convolution streams connected in parallel, and the fusion module. We hoped that by progressively training the model, the information from the representations from lower altitude images (which have higher resolutions) is shared with lower-resolution representations from images taken at higher altitudes.

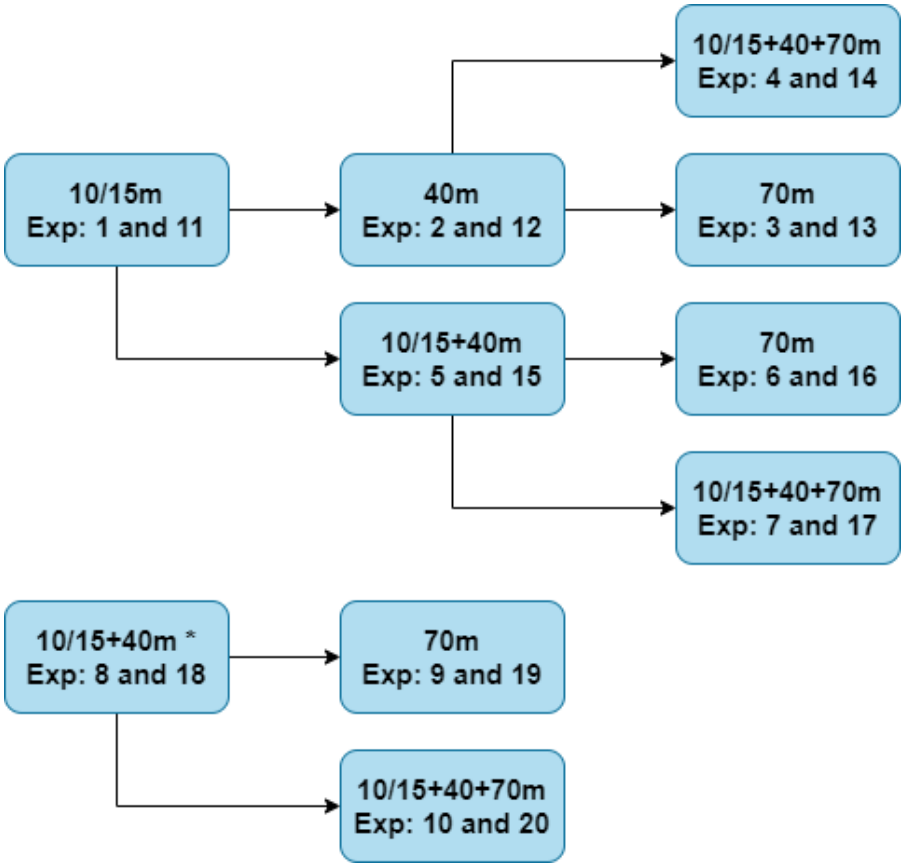


Figure 5.2: Experiments diagram. *Contains all images from both 10/15m and 40m.

5.2.2 Analyzing and Discussing Baseline Results

Before analyzing the results, please beware of the following: As we stated previously, we want to evaluate whether the model benefits from progressive training. Thus, we start training the model at lower altitudes (10/15m) and progressively increase the height. In some experiments, we use data from only one height, while in others, we also include some images from previous heights. This allows us to assess if the model benefits from having some images of previous configurations. The model was trained, validated and tested on different data sets, with no shared images between them. All the data sets configurations are detailed in table A.1, for each experiment.

Table 5.1: HRNet+OCR experiments results on the test data sets (short table).

Exp #	Hight	Pixel Acc	Class IoU	Producer's Acc.	User's Acc.	Pre-train	Pre-train weights	Train time
	10/15m	0.825	[0.810 0.302]	0.692	0.898			
3	40m	0.986	[0.985 0.819]	0.868	0.902	yes	exp_2	2h
	70m	0.993	[0.993 0.862]	0.913	0.901			
	10/15m	0.961	[0.949 0.850]	0.929	0.888			
4	40m	0.983	[0.981 0.788]	0.887	0.831	yes	exp_2	13h
	70m	0.984	[0.983 0.696]	0.866	0.751			
	10/15m	0.839	[0.824 0.342]	0.555	0.900			
6	40m	0.975	[0.973 0.678]	0.848	0.873	yes	exp_5	1h
	70m	0.993	[0.992 0.846]	0.907	0.865			
	10/15m	0.958	[0.946 0.835]	0.877	0.925			
7	40m	0.980	[0.979 0.749]	0.768	0.861	yes	exp_5	1h
	70m	0.987	[0.986 0.720]	0.768	0.850			
	10/15m	0.917	[0.900 0.661]	0.735	0.941			
9	40m	0.979	[0.977 0.738]	0.825	0.888	yes	exp_8	<1h
	70m	0.992	[0.991 0.823]	0.919	0.870			
	10/15m	0.952	[0.937 0.824]	0.946	0.887			
10	40m	0.984	[0.982 0.800]	0.895	0.831	yes	exp_8	6h
	70m	0.989	[0.988 0.774]	0.864	0.867			

Table A.2 contains the results obtained from the all experiments on the test data sets, and table 5.1 is a shorter table with only the results from the most important experiments. When the *Pre-train* column of the mentioned tables has the value *'no'*, it means the model was trained from scratch. Otherwise, it means the training was continued from the weights specified in the *Pre-train weights* column.

As mentioned in 2.4.2, using just pixel accuracy to evaluate a model's performance can be misleading if there is a class imbalance in the data set. As can be seen in table 5.1, that is precisely the case for our data and model performance. If we compare the pixel accuracy 5.12 to the user's accuracy 5.6 at higher altitudes, the value for pixel accuracy if disproportionately higher than the user's accuracy. Thus, we focus more on the class IoU 5.7 5.8 5.9, producer's accuracy, and user's accuracy 5.4 5.5 5.6 while evaluating the model's performance.

The experiments can be divided into six groups:

- $1- > 2- > 3$;
- $1- > 2- > 4$;
- $1- > 5- > 6$;
- $1- > 5- > 7$;
- $8- > 9$;
- $8- > 10$.

Considering these groups, the most important results are the ones from experiments 3, 4, 6, 7, 9, and 10. These are the last experiments of each group. Meaning these are the experiments after which the model was exposed to the whole data set.
Comparing the results of *exp 3* and *exp 4*, we can conclude the following: The resulting model from *exp 3* has better performance at high altitudes (40 and 70 m) 5.5 5.6 5.8 5.9, at the expense of sacrificing performance at lower attitudes when compared to the resulting model from *exp 4*. Inversely, the model from *exp 4* has better performance at lower altitudes 5.4 5.7 when compared with *exp 3* but worst performance at higher altitudes. These conclusions are supported by all metrics. The same conclusions can be made by comparing the results from *exp 6* and *exp 7*, and *exp 9* and *exp 10*, where *exp 7* and *exp 10* has better performance at lower altitudes, and *exp 6* and *exp 9* at higher altitudes. By comparing these results, we can already conclude that if the model has good performance at high altitudes it will have a lower performance at lower altitudes, and vice-versa.
Looking at the flow chart 5.2 presented previously, we can see that *exp 7* and *exp 10* where trained using progressive training, *exp 3* and *exp 9* used normal training (training one altitude at the time, without including images from previous altitudes), and the remaining experiments where trained using a combination of progressive and normal training. If we now compare the experiments 3, 4, 6, 7, 9, and 10 at 10/15 m, we clearly see that experiments 3, 4, 6 and 9 are the ones with the lowest performance. Meanwhile *exp 7* and *exp 10* are the best performing ones. However, at higher altitudes, despite now *exp 7* and *exp 10* performing worst that the previously mentioned experiments, the gap in performance is not as significant. Thus, we can also conclude that despite the loss of performance at higher altitudes, using progressive training results in an overall more robust model.
This being said, we want to have a model that has good performance at high altitudes, but that is also able to maintain good performance at lower attitudes. Thus, in our opinion the best models are the ones from *exp 7* and *exp 10*. They are not the best performing ones at 70 m, but they only lose slight performance when compared to the best at 70 m (*exp 3*), and they outperform all other models in the remaining altitudes.
In short, after gathering and analyzing the results from the experiments, we arrived at the following conclusions:

- Training the model without progressive training, results in a model with very low performance at low altitudes, but high performance at higher altitudes. On the other hand, training the model progressively results in a model with a lower performance at higher altitudes (but still comparable to the best performing models) and higher performance at low altitudes.
- Training a model with progressive training results in an overall more robust model compared to models trained normally or with partial progressive training.

5.2.3 Proposed Modifications to the HRNet+OCR Model

After establishing a baseline with HRNet [51], we focused on ways we could improve and adapt the model to our data/problem. One of our primary goals is to detect Ludwigia p. at higher altitudes (ideally on satellite imagery). Thus, we tried to find ways to simulate the appearance of satellite data using our data set. When compared to our data, satellite imagery has two distinct differences: it usually has a lower resolution and covers a broader area. So, we had to find ways to both lower our data's resolution and, if possible, enlarge the perceived Field of View (FoV) of the convolution layers.
To achieve the down-sampling of our data, we increased the *stride* value on the fusion module, for the connections between *stage* 1 and the remaining *stages*. This effectively reduces the perceived resolution of our images, making them more closely resemble higher altitude images. To enlarge the perceived FoV, we decided to use *atrous* convolutions (also known as dilated convolutions). This type of convolution allows enlarging the convolution's FoV by increasing the *dilation rate* value. If the *dilation rate* is set to zero, then we effectively have a "conventional" convolution layer. An added benefit of using dilated convolutions is that they save computation and memory costs, as increasing the value of the *dilation rate* allows for a larger receptive field which means viewing more data points. After implementing the aforementioned modifications, we expect the model to extract low-resolution features, resulting in better performance, especially in data captured at higher altitudes. We also expect the model to be more efficient due to the use of dilated convolutions and the down-sampling process.

5.2.4 Implementation Details

To implement the proposed modifications, we tried a series of different configurations. We experimented with different stride and dilation values and increased and decreased the number of stages. All of these trials were made using only train and validation data sets. A broad set of combinations of the mentioned configurations were tested. The main conclusions are the following:

- Increasing the number of stages resulted in a more complex and, as a result, a slower model. However, the increase in complexity did not translate into a comparable increase in performance;
- Decreasing the number of stages had the opposite effect. A faster model, but with considerably lower performance;
- Increasing the stride value of the fusion model on connections made form *stage* 1 resulted in a better performance at higher altitudes, but at the cost of a decrease at lower altitude performance;
- Increasing the value of dilation, reduced training time, and slightly increased performance at high altitudes, but at the cost of worst low altitude performance;

- Using different values of stride or dilation for each stage resulted in the worst performing configurations. For better results, stride and dilation values must be equal in all stages.

After establishing the behavior of the model for each configuration, we decided we had to compromise between two aspects of our model: (1) Increasing high altitude performance without significantly hindering the model's performance at low altitudes; (2) Decrease train time, without compromising the overall performance, especially at high altitudes. With these main observations in mind, we performed the following changes to the model (the architecture of the resulting model is represented in figure 5.3):

- We kept the same number of stages, as we did not see substantial benefits in either increasing or decreasing their number;
- We increased the *stride* value on the fusion module from 2 to 3 for all connections made between *stage* 1 and the remaining *stages*. All remaining connections kept the *stride* value at 2.
- We replaced the conventional convolution layers from *stages* 2 and 3, with dilated convolutions, with $dilation = 2$ and $padding = 2$. The convolution layers in *stages* 1 and 4 where kept unchanged.

After our initial tests (conducted using the train and validation data sets), we concluded that this is our best configuration. It seems to have achieved a balance between decreasing the training time with comparable performance (and sometimes even increased performance at higher altitudes, as is our goal). More extensive tests and results, are presented in the next section 5.2.5.

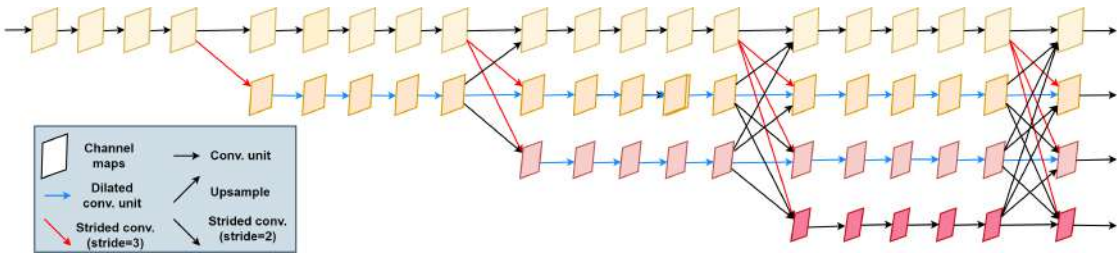


Figure 5.3: Architecture of our model, that is based on HRNet [51].

5.2.5 Tests and Results on the Modified HRNet+OCR

Like previously, we include tables 5.2, and A.3 with the results obtained from the experiments, on the test data sets. As for the datasets, we used the same ones as in the original model to ensure consistency.
The main conclusions for the modified model are homologous to the conclusions of the baseline results. However, we can see that the proposed changes to the model resulted in a significant decrease in training time, while maintaining comparable performance. The

Table 5.2: Modified HRNet+OCR experiments results on the test data sets (short table).

Exp #	Hight	Pixel Acc	Class IoU	Producer's Acc.	User's Acc.	Pre-train	Pre-train weights	Train time
	10/15m	0.851	[0.834 0.402]	0.426	0.880			
13	40m	0.986	[0.985 0.824]	0.886	0.922	yes	exp_12	1h
	70m	0.992	[0.991 0.896]	0.911	0.899			
	10/15m	0.951	[0.937 0.849]	0.888	0.902			
14	40m	0.981	[0.979 0.758]	0.823	0.906	yes	exp_12	8h
	70m	0.986	[0.985 0.710]	0.759	0.918			
	10/15m	0.843	[0.829 0.339]	0.343	0.976			
16	40m	0.985	[0.983 0.800]	0.840	0.944	yes	exp_15	1h
	70m	0.991	[0.990 0.878]	0.874	0.915			
	10/15m	0.950	[0.935 0.820]	0.959	0.850			
17	40m	0.987	[0.985 0.830]	0.899	0.916	yes	exp_15	1h
	70m	0.990	[0.989 0.769]	0.799	0.955			
	10/15m	0.842	[0.827 0.346]	0.355	0.934			
19	40m	0.979	[0.977 0.731]	0.783	0.917	yes	exp_18	<1h
	70m	0.989	[0.988 0.773]	0.854	0.891			
	10/15m	0.869	[0.934 0.803]	0.890	0.892			
20	40m	0.848	[0.974 0.720]	0.836	0.839	yes	exp_18	1h
	70m	0.821	[0.980 0.662]	0.840	0.758			

overall training time dropped from 130 h to 57 h A.1, representing 43.9% less training time, which means that we have comparable performance at less than half the training time.

The user's accuracy 5.4 5.5 5.6 have remained more or less the same for experiments 14, 17, and 20, but significantly doped in experiments 13, 16 and 19. Note, that these are the experiments with the lowest performance in both class IoU 5.7 and pixel accuracy 5.10 at 10/15 m. The producer's accuracy has also remained comparable, having increased in some experiments and decreased in others.

During the training phase, both *exp 19* and *exp 20* did not behave as expected. We ran the experiments multiple times, but the training always stopped much earlier than expected, while still having high validation loss. As a result, we got mixed results for the class IoU. While having generally increased in experiments 13, 14, 16, and 17, especially at higher altitudes 5.8 5.9, it decreased for *exp 19* and *exp 20*. These observations are also true for the pixel accuracy metric 5.10 5.11 5.12.

We believe that this is due to the fact that *exp 19* and *exp 20* use the pre-trained weights from *exp 18*. Experiments 13, 14, 16, and 17 all start from *exp 11* that only trains with the 10/15 data set. While experiments 13, 14, 16, and 17 then have an intermediate step at either *exp 12* or *exp 15* (where the model learns features at 40 m), experiments 19 and 20 stem from *exp 18*, that does not use pre-trained weights. Furthermore *exp 18* used a training set that consists of both the training data sets of 10/15 and 40 m (more details about the data set composition can be seen at A.1).

Because we changed the architecture of the model, it may have unpredictably affected: (1) the way the high-to-low convolutions and fusion module combine the information between the parallel streams; (2) the OCR module. Because *exp 18* is the first training, and

it uses both 10/15 and 40 m images, the model may have trouble learning features, due to the excess of representations. On the other hand, experiments that stem from *exp 11*, only introduce one new height at the time, which may be simplifying the representations, allowing the model to learn each representations separately. After carefully analyzing all the possible reasons, this was the one that made the most sense in justifying why the experiments 18, 19, and 20 have such an unexpected behavior, compared to experiments 8, 9, and 10.

In short, the resulting model is better suited to our data and application. Our model is faster when it comes to training time while maintaining similar performance and has a slight performance increase in high-altitude images.

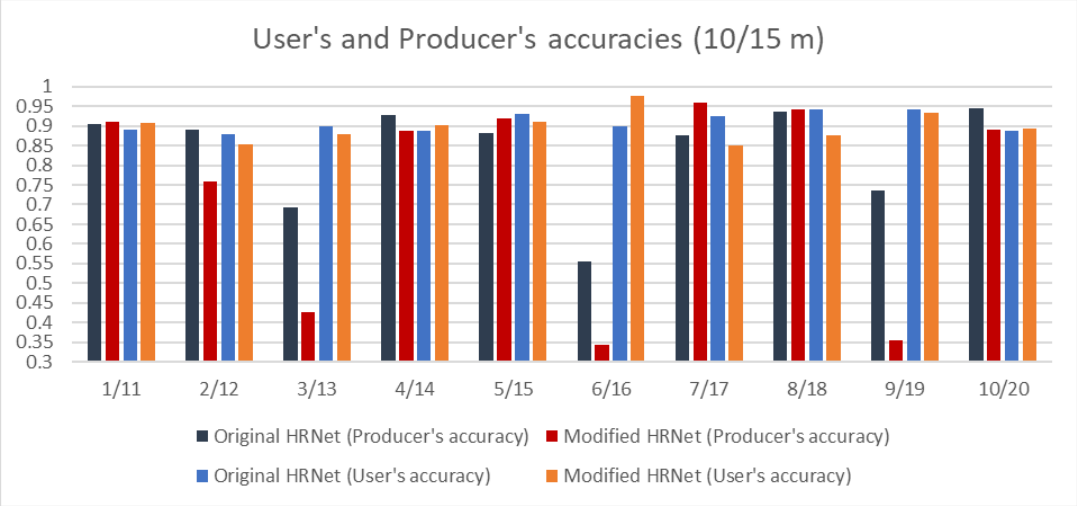


Figure 5.4: User's and producer's accuracies at 10/15 m.

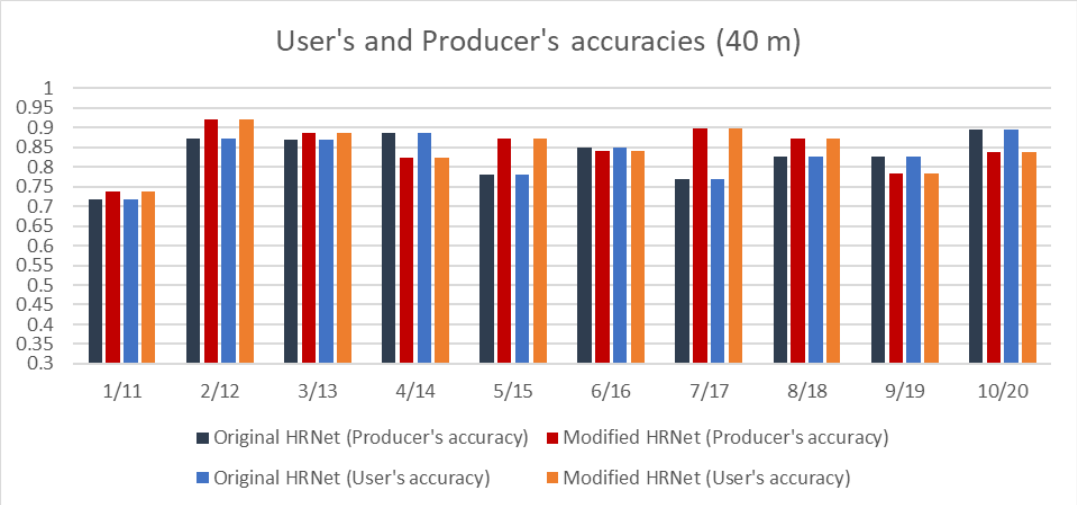


Figure 5.5: User's and producer's accuracies at 40 m.

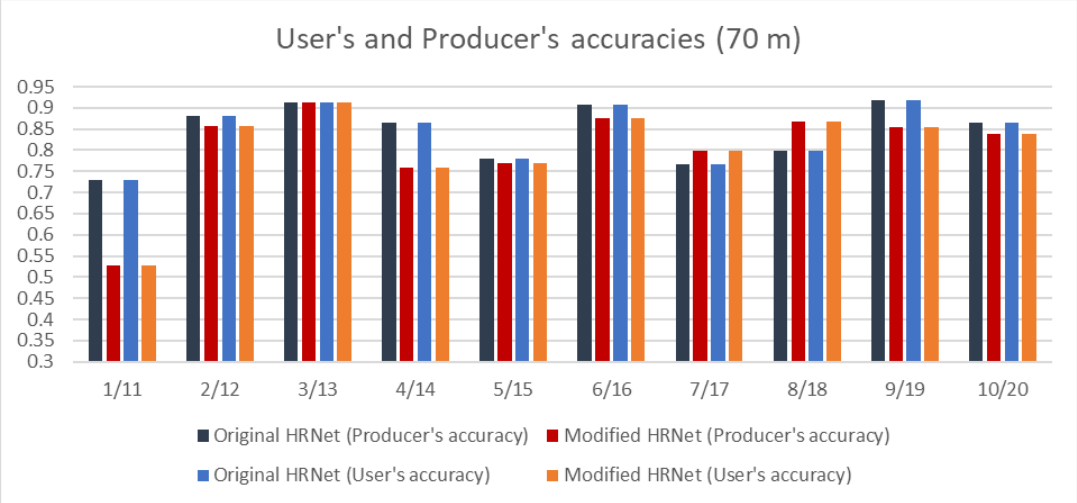


Figure 5.6: User's and producer's accuracies at 70 m.

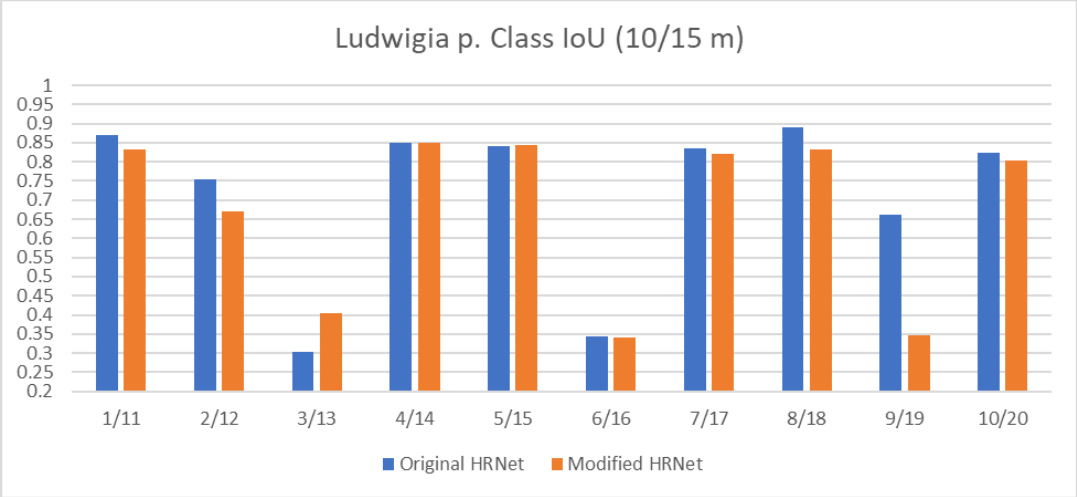


Figure 5.7: Ludwigia p. Class IoU at 10/15m.

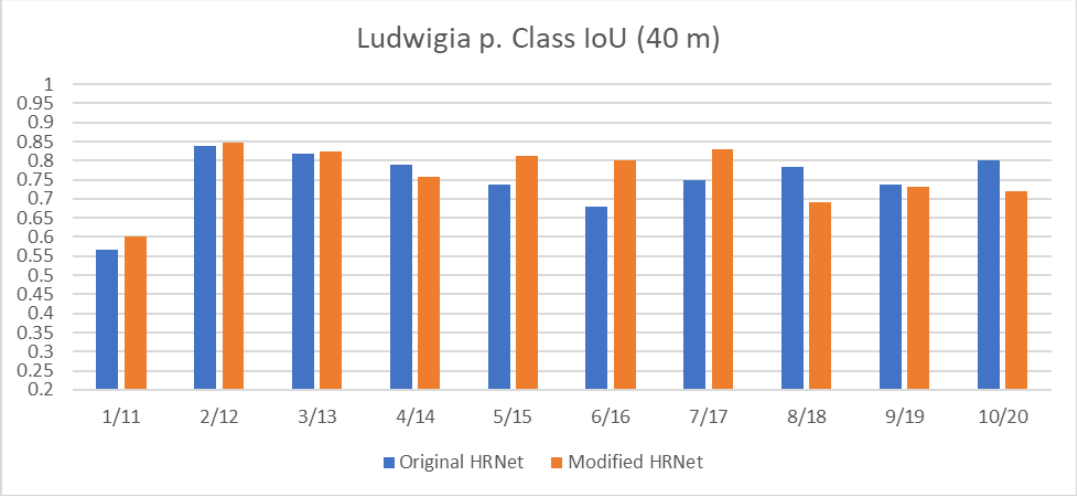


Figure 5.8: Ludwigia p. Class IoU at 40m.

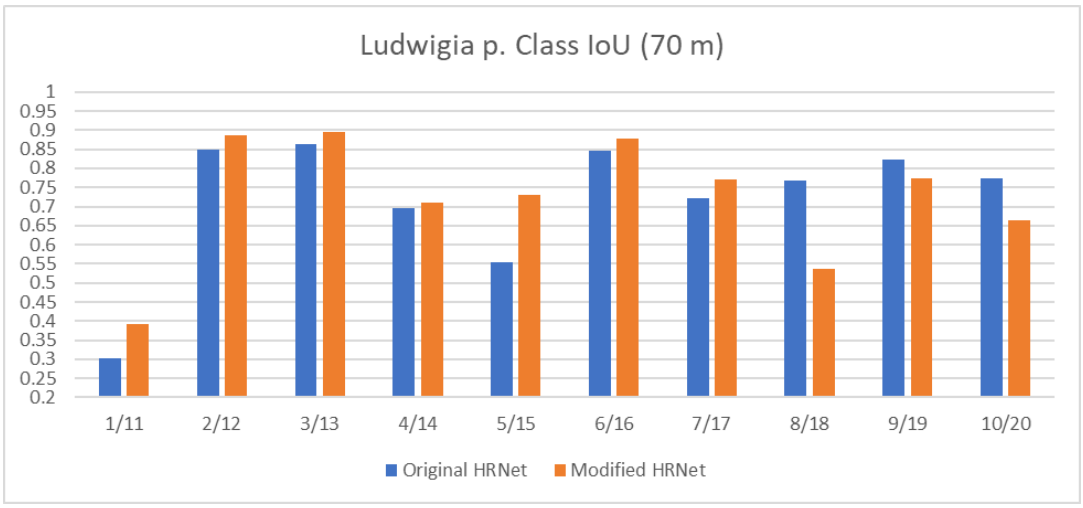


Figure 5.9: Ludwigia p. Class IoU at 70m.

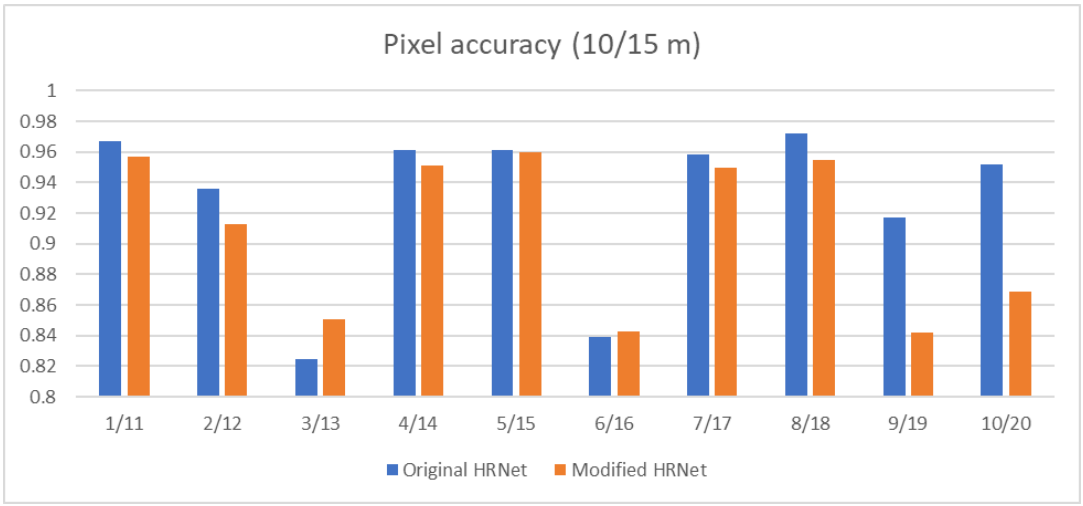


Figure 5.10: Pixel accuracies at 10/15m.

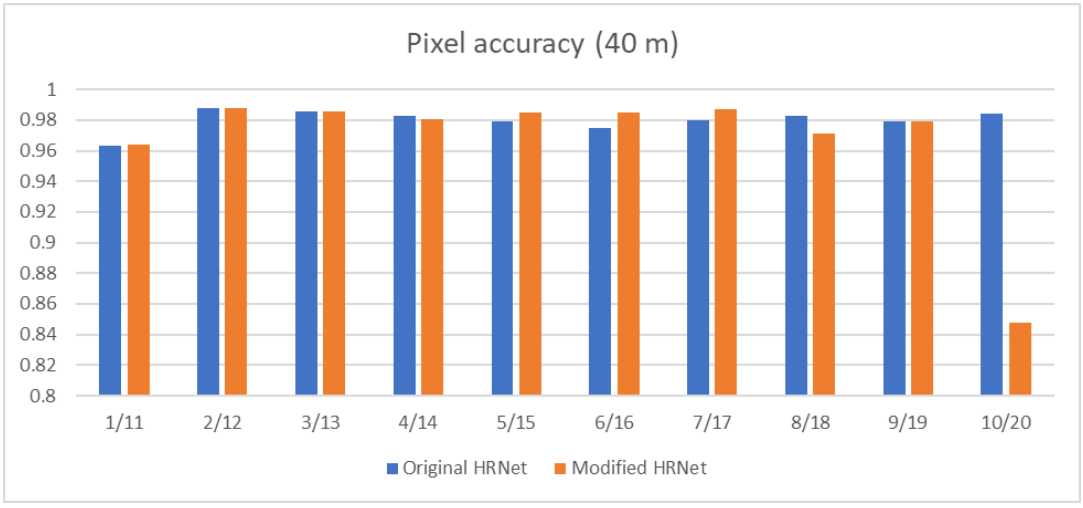


Figure 5.11: Pixel accuracies at 40m.

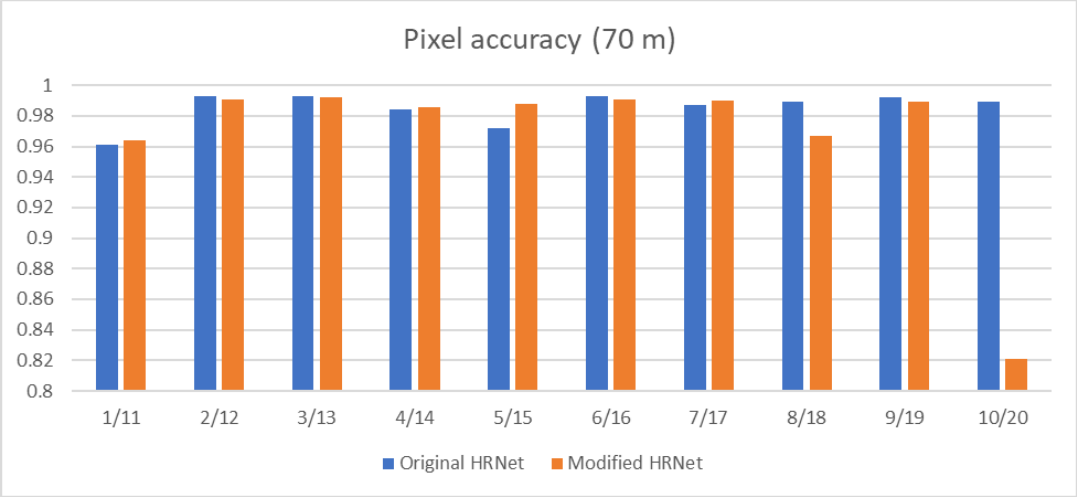


Figure 5.12: Pixel accuracies at 70m.

5.2.6 Comparison and Discussion of Results

As stated previously, the modified model has comparable performance at less than half the training time. These values confirm that our proposed modifications indeed resulted in the predicted outcome. This is especially true for high-altitude images, which was one of the main goals of the modifications.
Furthermore, by looking at both the user's and producer's accuracy, we can conclude that our model produces high-reliability maps and has the tendency to output more false positives than false negatives. This is great because it is better to have a model that occasionally identifies other species as Ludwigia p., then having a model that sometimes fails to identify the IAS. In real-world scenarios, not identifying a single strand of Ludwigia p. may lead to identifying a site falsely as not being infested. Due to the capability of the IAS spreading rapidly from one single seed, this can lead to a severe infestation in the future. Ludwigia p. reproduces aggressively, and in a brief period, one misclassified plant can lead to a wholly covered body of water.
As for the best overall model, we consider it to be the one from *exp 17*. It has a combined train time of just 24 h (compared to the 36 h of the base model) and has excellent performance at both high and low altitudes. The best model for high altitude is the model from *exp 3*, as it is the one with the best performance results for 70 m.

As for the comparison with other authors, we compare our model's performance to the performance achieved by Bolch in [9]. His work, setup, and objectives are very similar to ours, and he also assesses the performance on water primrose (which is an alternate designation for Ludwigia spp. Like already mentioned in 3.1.1, it is very similar to Ludwigia p.). He achieved a producer's accuracy of 100% on water primrose and a user's accuracy of just 50%. Our best model at 70 m (the highest altitude on our data set), achieved a producer's accuracy of 79.9% (an 20.1% decrease compared to [9]), and a user's accuracy

of 95.5% (an increase of 45.5%). We have to keep in mind that the sensor used in [9] is superior to ours and has more bands (their sensor has 270 bands, while our sensor only has 5 bands).

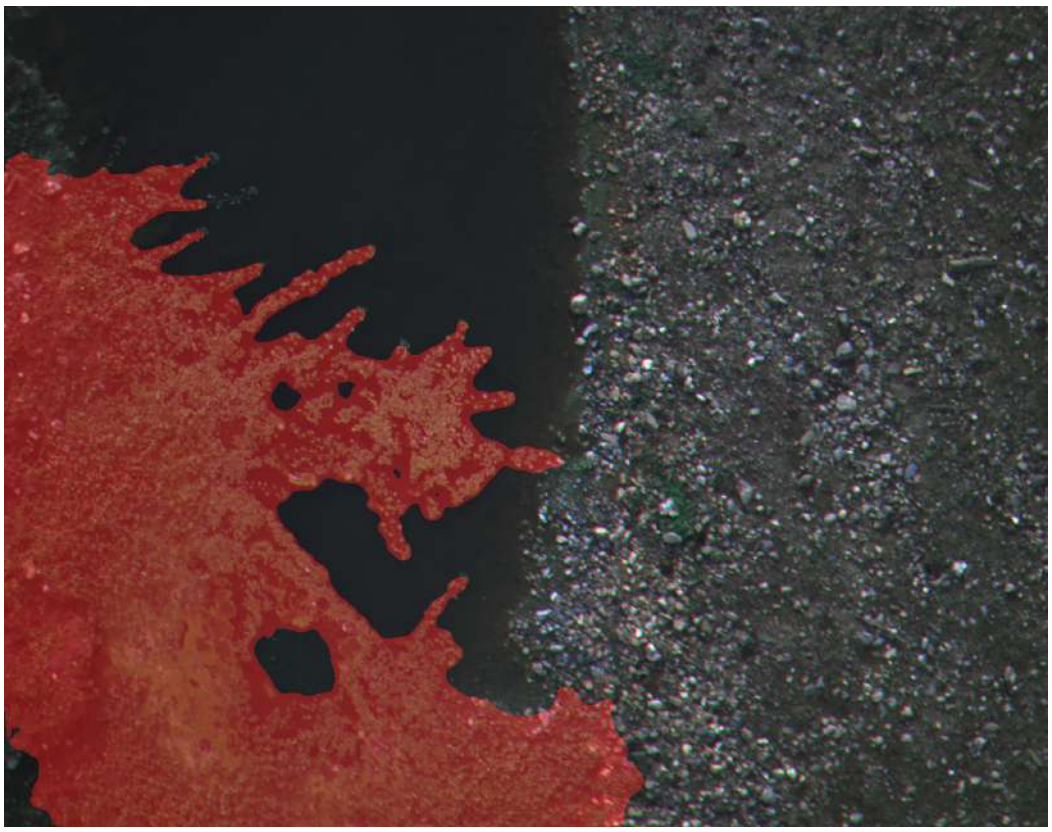
5.2.7 Analyzing the Model's Output

After analyzing and comparing quantitative results, we shifted our focus to analyzing the actual outputs of the model. Although it is imperative to evaluate the model using quantitative results, a visual analysis of the predictions allows us to understand the scenarios where the model is and is not accurate. These observations will help us address the model's weaknesses during future work. After looking at various output predictions and overlapping them to their respective images, we were able to take a few conclusions:

- As already proven by the quantitative results, the model is very accurate in the predictions it makes;
- Sometimes the model has some difficulty identifying smaller stems of the species, especially from images captured at high altitudes;
- In cases where there are small gaps in the mantle, the model usually classifies the gap as Ludwigia peploides. This can be seen in figures 5.14, and 5.16;
- Occasionally, the model identifies some surrounding vegetation as Ludwigia p. This is especially the case for surrounding bramble bushes, as can be seen in figure 5.15;
- The model sometimes struggles to identify Ludwigia in darker spots. One example of this can be seen in the top left corner of figure 5.13, where the model failed to identify the darkest part.

Considering the quantitative results and our observations, we further conclude that the model has good performance. It occasionally misses some spots, especially in lower light conditions, and smaller stems at high altitudes. Also, it tends to mark small gaps in the mantle as being the targeted species and some surrounding native species. In future work, we will address these issues by improving the model itself and expanding our data set to cover more light and atmospheric conditions.

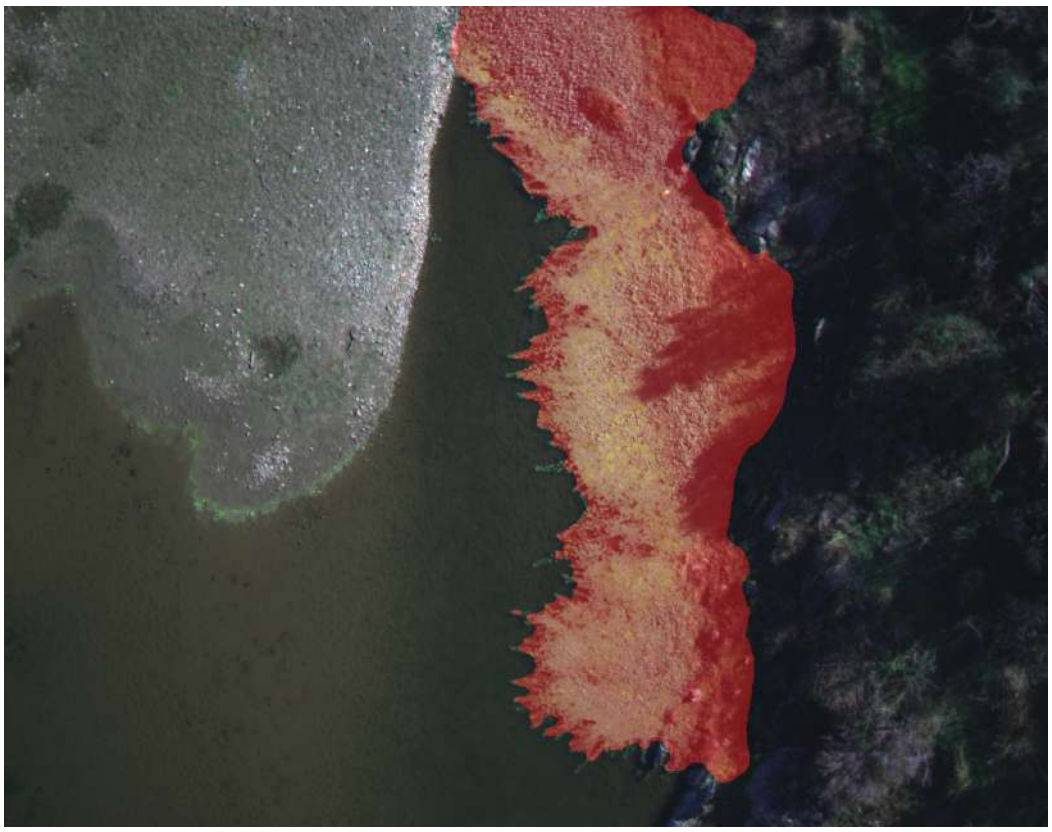
(a) (b)



(c)

Figure 5.13: Ground truth (a), model prediction (b), and prediction overlapped on image (c). Image taken at 15m.

(a) (b)

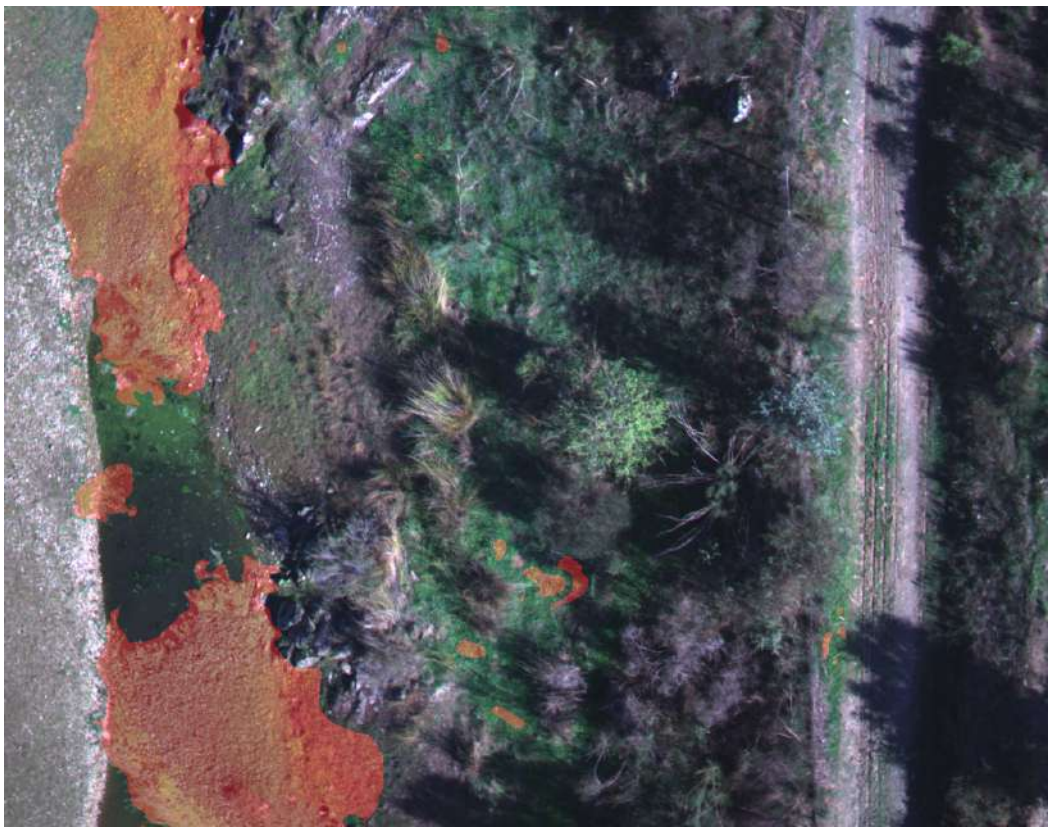


(c)

Figure 5.14: Ground truth (a), model prediction (b), and prediction overlapped on image (c). Image taken at 40m.

(a)

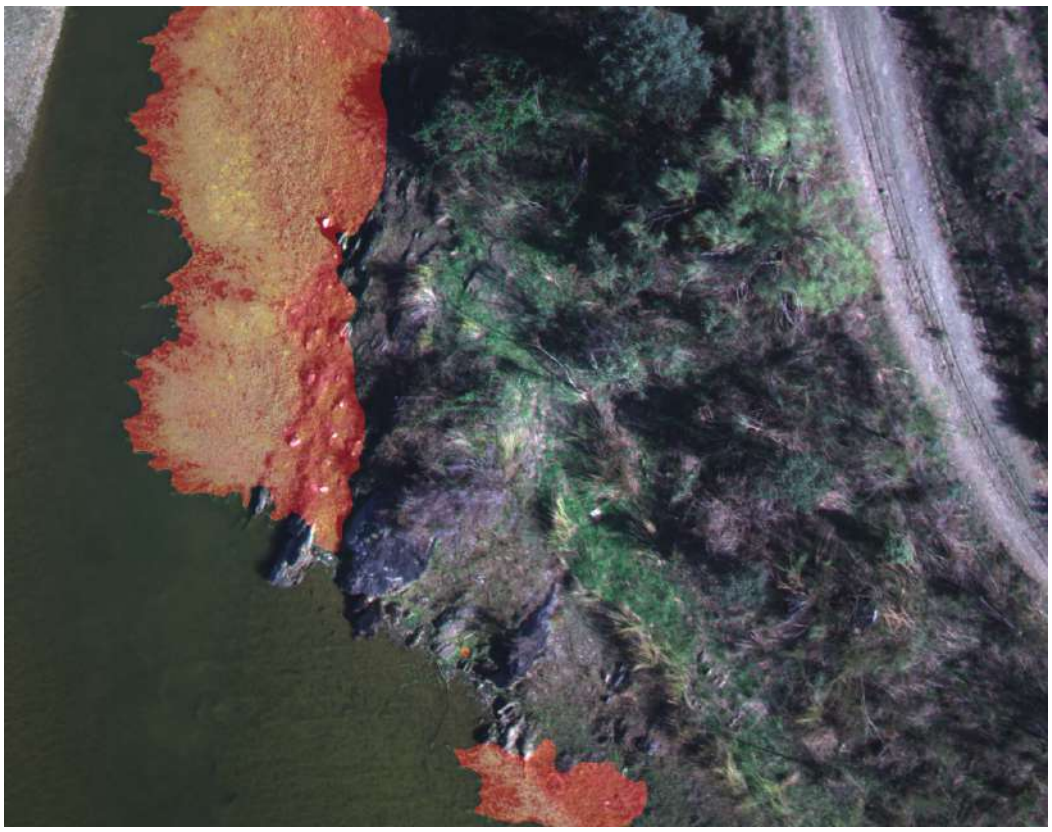
(b)



(c)

Figure 5.15: Ground truth (a), model prediction (b), and prediction overlapped on image (c). Image taken at 40m.

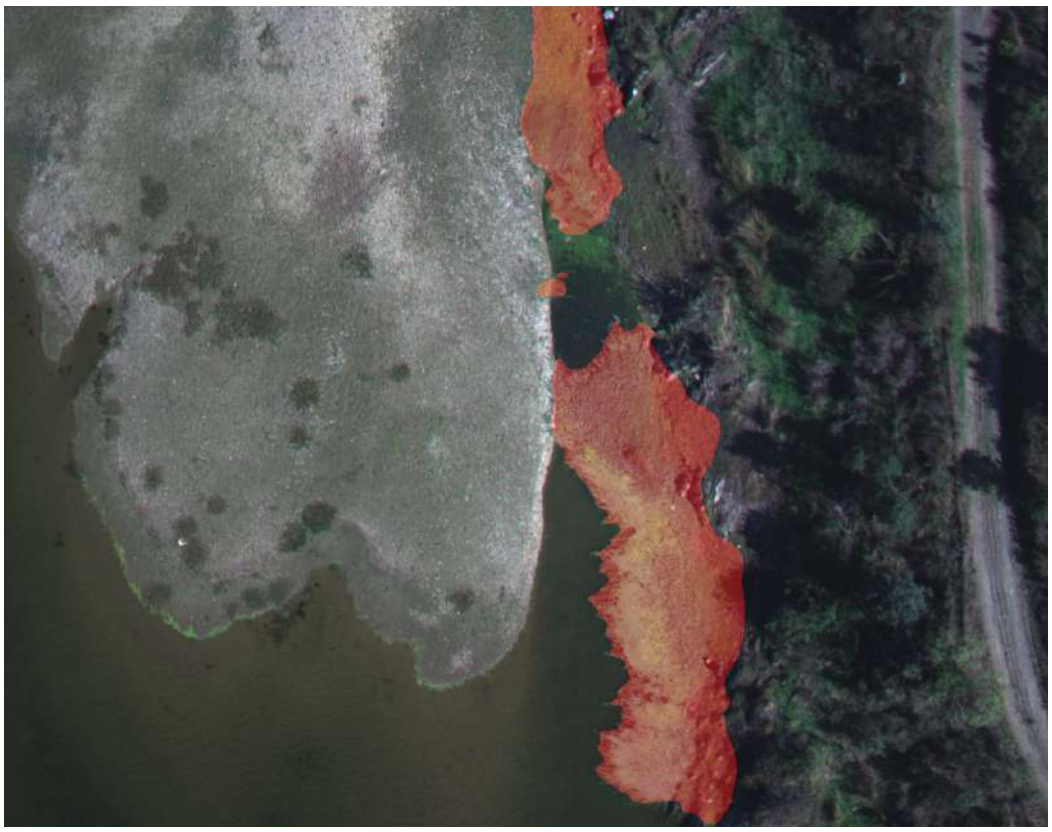
(a) (b)



(c)

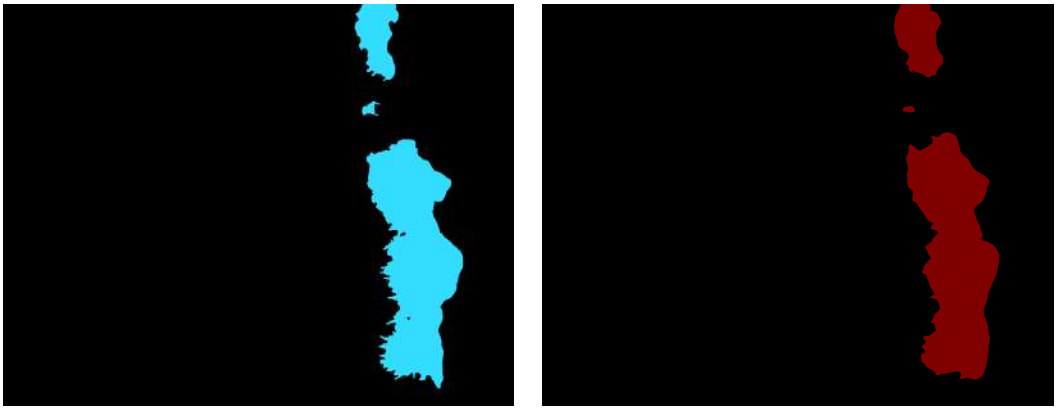
Figure 5.16: Ground truth (a), model prediction (b), and prediction overlapped on image (c). Image taken at 40m.

(a) (b)

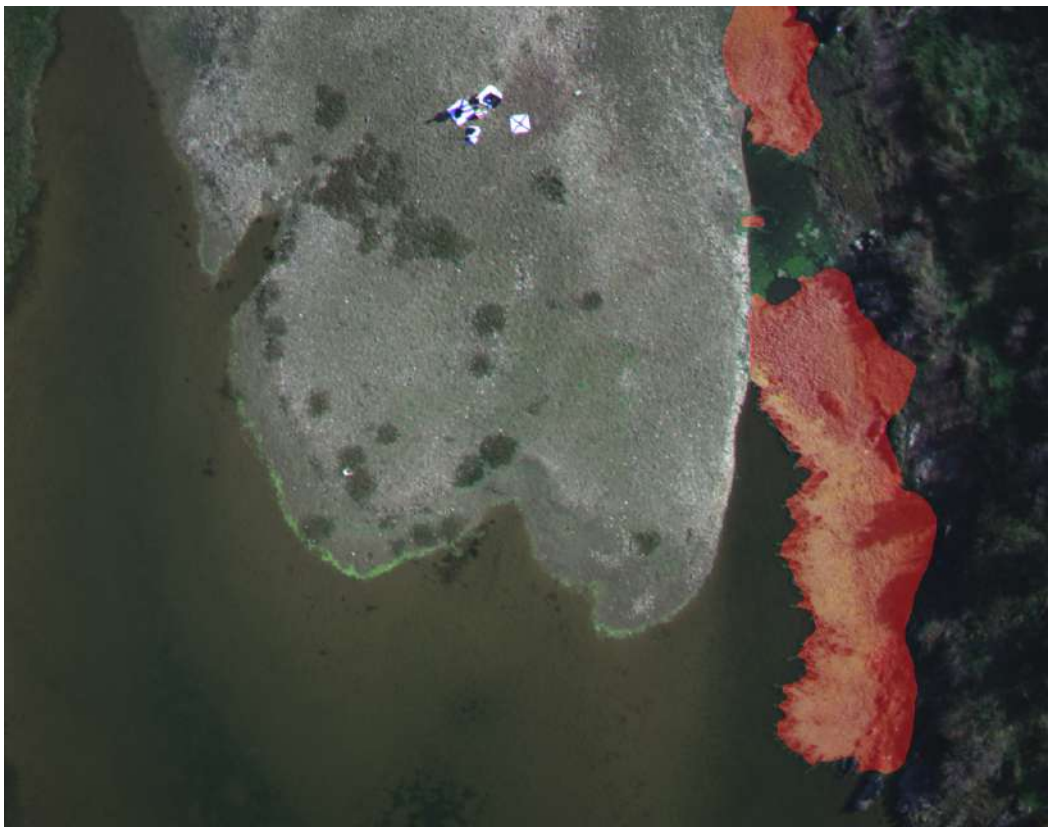


(c)

Figure 5.17: Ground truth (a), model prediction (b), and prediction overlapped on image (c). Image taken at 70m.



(a) (b)



(c)

Figure 5.18: Ground truth (a), model prediction (b), and prediction overlapped on image (c). Image taken at 70m.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

We started our work by analyzing essential concepts regarding RS and detection of IAS. This was a vital step that helped us to understand the problem better and find viable ways to solve it.

We studied the platforms and sensors available and used by other authors. We first experimented with satellite images, but the one we were able to acquire had very low resolution. Given that the infestation was still in its early stages (thus, the spread of Ludwigia p. was not significant), it was impossible to identify the IAS.

Thus, we had to carefully compare other available sensors and platforms that would be the most adequate, considering our budget restrictions. After reading the work done by Bolch in [9], we were inspired to use drone-mounted imaging. Given the nature of UAVs, they are highly flexible, allow a broad control of both the configurations and periods in which the data was captured, and are the most affordable option to acquire and maintain. Unfortunately, despite being the cheapest option, some drone and sensor configurations can still cost upwards of 100000 €. Given the limited budget for this project, we had to compromise and choose a less capable sensor and platform than those used by other authors [9, 31]. Despite having fewer bands (only five), we proved that it was still possible to isolate the spectral signature of Ludwigia p., using our setup.

After analyzing the models used by other authors and state-of-the-art segmentation models, we decided that we would use semantic segmentation models to identify the targeted species. By using semantic segmentation models, we were able to leverage both the capabilities of these models to recognize objects and the multispectral nature of our data. Before being able to start training and testing models, we need first to have a data set. Unfortunately, we could not find any publicly available data sets and had to build our own.

We visited the study site twice and captured data at multiple altitudes, periods of the day, and atmospheric conditions. We wanted to make sure that our data set was as representative as possible and contained variations of the atmospheric conditions. After collecting the necessary images, we started the processing steps of the data set. Our drone already does some of the steps automatically, namely correcting the images according to light conditions. However, because the five bands are captured as separate images, we needed to join them to create a single image. Given the positioning of the sensors and manufacturing tolerances in the sensors lenses, we had first to align the images. After the images were aligned, they were joined as a single image that would be the ones used to create the data

set. Once the data set was created, we needed to complete it with the corresponding labels for the segmentation models. Once annotated, we could use our data set to train and test semantic segmentation models. We first used a preexisting model (HRNet [51]). We had to modify the model, especially the input layers of the model, for it to be able to accommodate our data. After the necessary changes were made, we achieved excellent results. Later, we proposed a set of modifications to the model. The goal was to increase its performance, especially at high altitudes, and reduce the training time. As shown previously, we met our goal. The resulting model needs considerably less training time while increasing performance in the scenarios we want. Comparing our results to the ones achieved by other authors, we have comparable performance using simpler data.
Overall, we consider our work a success. We achieved our main goal: remotely detecting the presence of Ludwigia peploides using drone-mounted multispectral data. During the process, we created a data set and proved that it is possible to accurately identify IAS without the need for state-of-the-art.

6.2 Future Work

As for future work, we have four main goals:

1. Be able to detect Ludwigia peploides using satellite images. This would allow proper remote detection, only deploying drones in sites where the targeted species have been identified. The use of drones would be to have a more detailed view of the area;

2. Expand our data set to cover more diverse light and atmospheric conditions;

3. We also plan on extending this work to other species. Plans are already being made for a sister project to monitor forests;

4. Finally, we will further improve the model to address some previously stated issues. Ideally, we want to create a custom model from scratch based on what we learned from this work.

Bibliography

[1] Decreto-Lei 92/2019, 2019-07-10 - DRE. Last accessed in: 2021-10-02. [Online]. Available at https://dre.pt/dre/detalhe/decreto-lei/92-2019-123025739. 2

[2] EHS - Hyperspectral Remote Sensing Scenes. Last accessed in: 2021-12-10. [Online]. Available at https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes. xvii, xix, 36, 37, 38

[3] EUR-Lex - 32014R1143 - EN - EUR-Lex. Last accessed in: 2021-10-02. [Online]. Available at https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32014R1143. 2

[4] Humboldt State University. Last accessed in: 2021-11-15. [Online]. Available at http://gsp.humboldt.edu/olm/Courses/GSP_216/lessons/accuracy/metrics.html. xvii, 13

[5] LudVision article preprint submitted to Remote Sensing of Environment. Last accessed in: 2022-06-27. [Online]. Available at https://github.com/antoniojmabreu/LudVision/blob/main/Article.pdf. 2

[6] John B. Adams, Donald E. Sabol, Valerie Kapos, Raimundo Almeida Filho, Dar A. Roberts, Milton O. Smith, and Alan R. Gillespie. Classification of multispectral images based on fractions of endmembers: Application to land-cover change in the Brazilian Amazon. *Remote Sensing of Environment*, 52(2):137–154, may 1995. 32

[7] Joshua L. Bandfield, Philip R. Christensen, and Michael D. Smith. Spectral data set factor analysis and end-member recovery: Application to analysis of Martian atmospheric particulates. *Journal of Geophysical Research: Planets*, 105(E4):9573–9587, apr 2000. 7

[8] Jackson Baron and D. J. Hill. Monitoring grassland invasion by spotted knapweed (Centaurea maculosa) with RPAS-acquired multispectral imagery. *Remote Sensing of Environment*, 249, nov 2020.

[9] Erik A. Bolch, Erin L. Hestir, and Shruti Khanna. Performance and Feasibility of Drone-Mounted Imaging Spectroscopy for Invasive Aquatic Vegetation Detection. *Remote Sensing 2021, Vol. 13, Page 582*, 13(4):582, feb 2021. xix, 1, 5, 27, 28, 29, 30, 31, 34, 52, 63, 64, 71

[10] Erik A. Bolch, Maria J. Santos, Christiana Ade, Shruti Khanna, Nicholas T. Basinger, Martin O. Reader, and Erin L. Hestir. Remote detection of invasive alien species. *Remote Sensing of Plant Biodiversity*, pages 267–307, jan 2020. 1, 5, 6, 7, 8

[11] Bethany A. Bradley. Remote detection of invasive plants: A review of spectral, textural and phenological approaches. *Biological Invasions*, 16(7):1411–1425, oct 2014. xvii, 1, 5, 9

[12] Tanmay Chakraborty and Utkarsh Trehan. SpectralNET: Exploring Spatial-Spectral WaveletCNN for Hyperspectral Image Classification. apr 2021. 24

[13] Liang Chieh Chen, Maxwell D. Collins, Yukun Zhu, George Papandreou, Barret Zoph, Florian Schroff, Hartwig Adam, and Jonathon Shlens. Searching for Efficient Multi-Scale Architectures for Dense Image Prediction. *Advances in Neural Information Processing Systems*, 2018-Decem:8699–8710, sep 2018.

[14] Liang Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, jun 2016. xvii, 18, 19, 21

[15] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking Atrous Convolution for Semantic Image Segmentation. jun 2017. 18, 19, 22

[16] Liang Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L. Yuille. Attention to Scale: Scale-aware Semantic Image Segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem:3640–3649, nov 2015.

[17] Liang Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11211 LNCS:833–851, feb 2018. xvii, 18, 19, 20

[18] François Chollet. Xception: Deep Learning with Depthwise Separable Convolutions. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua:1800–1807, oct 2016. 20

[19] Jie Dai, Dar A. Roberts, Doug A. Stow, Li An, Sharon J. Hall, Scott T. Yabiku, and Phaedon C. Kyriakidis. Mapping understory invasive plant species with field and remotely sensed data in Chitwan, Nepal. *Remote Sensing of Environment*, 250, dec 2020.

[20] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable Convolutional Networks. *Proceedings of the IEEE International Conference on Computer Vision*, 2017-Octob:764–773, mar 2017. 20

[21] Sophie Dandelot, Christine Robles, Nicolas Pech, Arlette Cazaubon, and Régine Verlaque. Allelopathic potential of two invasive alien Ludwigia spp. *Aquatic Botany*, 88(4):311–316, may 2008. 1

[22] Belinda Gallardo, Miguel Clavero, Marta I. Sánchez, and Montserrat Vilà. Global ecological impacts of invasive species in aquatic ecosystems. *Global Change Biology*, 22(1):151–163, jan 2016. 1

[23] Hamed Gholizadeh, Michael S. Friedman, Nicholas A. McMillan, William M. Hammond, Kianoosh Hassani, Aisha V. Sams, Makyla D. Charles, De Andre R. Garrett, Omkar Joshi, Robert G. Hamilton, Samuel D. Fuhlendorf, Amy M. Trowbridge, and Henry D. Adams. Mapping invasive alien species in grassland ecosystems using airborne imaging spectroscopy and remotely observable vegetation functional traits. *Remote Sensing of Environment*, 271, mar 2022.

[24] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, 2017-Octob:2980–2988, dec 2017.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem:770–778, dec 2015. 20

[26] Kate S. He, Duccio Rocchini, Markus Neteler, and Harini Nagendra. Benefits of hyperspectral remote sensing for tracking plant invasions. *Diversity and Distributions*, 17(3):381–392, may 2011. 1, 5

[27] Erin L. Hestir, Shruti Khanna, Margaret E. Andrew, Maria J. Santos, Joshua H. Viers, Jonathan A. Greenberg, Sepalika S. Rajapakse, and Susan L. Ustin. Identification of invasive vegetation using hyperspectral remote sensing in the California Delta ecosystem. *Remote Sensing of Environment*, 112(11):4034–4047, nov 2008. 32

[28] Akira Hirano, Marguerite Madden, and Roy Welch. Hyperspectral image data for mapping wetland vegetation. *Wetlands 2003 23:2*, 23(2):436–448, 2003. 32

[29] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. apr 2017. 20

[30] Teja Kattenborn, Javier Lopatin, Michael Förster, Andreas Christian Braun, and Fabian Ewald Fassnacht. UAV data as alternative to field sampling to map woody invasive species based on combined Sentinel-1 and Sentinel-2 data. *Remote Sensing of Environment*, 227:61–73, jun 2019.

[31] Shruti Khanna, Maria J. Santos, Jennifer D. Boyer, Kristen D. Shapiro, Joaquim Bellvert, and Susan L. Ustin. Water primrose invasion changes successional pathways in an estuarine ecosystem. *Ecosphere*, 9(9):e02418, sep 2018. xvii, xix, 1, 5, 27, 30, 31, 32, 33, 34, 52, 71

[32] Shruti Khanna, Maria J. Santos, Erin L. Hestir, and Susan L. Ustin. Plant community dynamics relative to the changing distribution of a highly invasive species, Eichhornia crassipes: A remote sensing perspective. *Biological Invasions*, 14(3):717–733, mar 2012. 33

[33] Shruti Khanna, Maria J. Santos, Susan L. Ustin, and Paul J. Haverkamp. An integrated approach to a biophysiologically based classification of floating aquatic macrophytes. *https://doi.org/10.1080/01431160903505328*, 32(4):1067–1094, 2011. 6, 31

[34] Philipp Krähenbühl and Vladlen Koltun. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011*, oct 2012.

[35] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25, 2012. 20

[36] Lukas W. Lehnert, Hanna Meyer, Wolfgang A. Obermeier, Brenner Silva, Bianca Regeling, Boris Thies, and Jörg Bendix. Hyperspectral data analysis in R: The hsdar package. *Journal of Statistical Software*, 89, 2019.

[37] Anushree Malik. Environmental challenge vis a vis opportunity: The case of water hyacinth. *Environment International*, 33(1):122–138, jan 2007. 1

[38] Stefan Nehring and Detlef Kolthoff. The invasive water primrose Ludwigia grandiflora (Michaux) Greuter & Burdet (Spermatophyta: Onagraceae) in Germany: First record and ecological risk assessment. *Aquatic Invasions*, 6(1):83–89, 2011. 1

[39] R Gll Pontlus. Quantification Error Versus Location Error in Comparison of Categorical Maps.

[40] Gillian S.L. Rowan and Margaret Kalacska. A review of remote sensing of submerged aquatic vegetation for non-specialists. *Remote Sensing*, 13(4):1–50, feb 2021.

[41] Swalpa Kumar Roy, Shiv Ram Dubey, Subhrasankar Chatterjee, and Bidyut Baran Chaudhuri. FuSENet: Fused squeeze-and-excitation network for spectral-spatial hyperspectral image classification. *IET Image Processing*, 14(8):1653–1661, jun 2020. 24

[42] Swalpa Kumar Roy, Gopal Krishna, Shiv Ram Dubey, and Bidyut B. Chaudhuri. HybridSN: Exploring 3D-2D CNN Feature Hierarchy for Hyperspectral Image Classification. *IEEE Geoscience and Remote Sensing Letters*, 17(2):277–281, feb 2019.

[43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, Li Fei-Fei, O Russakovsky, J Deng, H Su, J Krause, S Satheesh, S Ma, Z Huang, A Karpathy, A Khosla, M Bernstein, A C Berg, and L Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, sep 2014. 20

[44] Maria J. Santos, Erin L. Hestir, Shruti Khanna, and Susan L. Ustin. Image spectroscopy and stable isotopes elucidate functional dissimilarity between native and nonnative plant species in the aquatic environment. *New Phytologist*, 193(3):683–695, feb 2012. 6, 32

[45] Ben Somers and Gregory P. Asner. Multi-temporal hyperspectral mixture analysis and feature selection for invasive species mapping in rainforests. *Remote Sensing of Environment*, 136:14–27, sep 2013.

[46] Iris Stiers, Nicolas Crohain, Guy Josens, and Ludwig Triest. Impact of three aquatic invasive species on native plants and macroinvertebrates in temperate ponds. *Biological Invasions 2011 13:12*, 13(12):2715–2726, feb 2011. 1

[47] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:10691–10700, may 2019.

[48] Lise Thouvenot, Jacques Haury, and Gabrielle Thiebaut. A success story: water primroses, aquatic plant pests. *Aquatic Conservation: Marine and Freshwater Ecosystems*, 23(5):790–803, oct 2013. 1

[49] Viktor R. Tóth, Paolo Villa, Monica Pinardi, and Mariano Bresciani. Aspects of Invasiveness of Ludwigia and Nelumbo in Shallow Temperate Fluvial Lakes. *Frontiers in Plant Science*, 0:647, apr 2019. 5

[50] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, MARCH 2020 Deep High-Resolution Representation Learning for Visual Recognition.

[51] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep High-Resolution Representation Learning for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3349–3364, aug 2019. xvii, xviii, 20, 21, 22, 51, 52, 53, 57, 58, 72

[52] Peter J. Weisberg, Thomas E. Dilts, Jonathan A. Greenberg, Kerri N. Johnson, Henry Pai, Chris Sladek, Christopher Kratt, Scott W. Tyler, and Alice Ready. Phenology-

based classification of invasive annual grasses to the species level. *Remote Sensing of Environment*, 263, sep 2021.

[53] Tian Zhu Xiang, Gui Song Xia, and Liangpei Zhang. Mini-Unmanned Aerial Vehicle-Based Remote Sensing: Techniques, applications, and prospects. *IEEE Geoscience and Remote Sensing Magazine*, 7(3):29–63, sep 2019.

[54] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. may 2021.

[55] Huang Yao, Rongjun Qin, and Xiaoyu Chen. Unmanned Aerial Vehicle for Remote Sensing Applications—A Review. *Remote Sensing 2019, Vol. 11, Page 1443*, 11(12):1443, jun 2019. 5, 39

[56] Fisher Yu and Vladlen Koltun. Multi-Scale Context Aggregation by Dilated Convolutions. *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, nov 2015.

[57] Yuhui Yuan, Xiaokang Chen, Xilin Chen, and Jingdong Wang. Segmentation Transformer: Object-Contextual Representations for Semantic Segmentation. *arXiv*, pages 1–23, sep 2019. xvii, 22, 23, 52

[58] Yuhui Yuan, Xiaokang Chen, Xilin Chen, and Jingdong Wang. Segmentation Transformer: Object-Contextual Representations for Semantic Segmentation. *arXiv*, pages 1–23, sep 2019.

[59] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, Mu Li, Alexander Smola, and Uc Davis. ResNeSt: Split-Attention Networks. apr 2020.

[60] Qiqi Zhu, Weihuan Deng, Zhuo Zheng, Yanfei Zhong, Qingfeng Guan, Weihua Lin, Liangpei Zhang, and Deren Li. A Spectral-Spatial-Dependent Global Learning Framework for Insufficient and Imbalanced Hyperspectral Image Classification. *IEEE Transactions on Cybernetics*, may 2021. 23

Appendix A

A.1 Example of Band Alignment

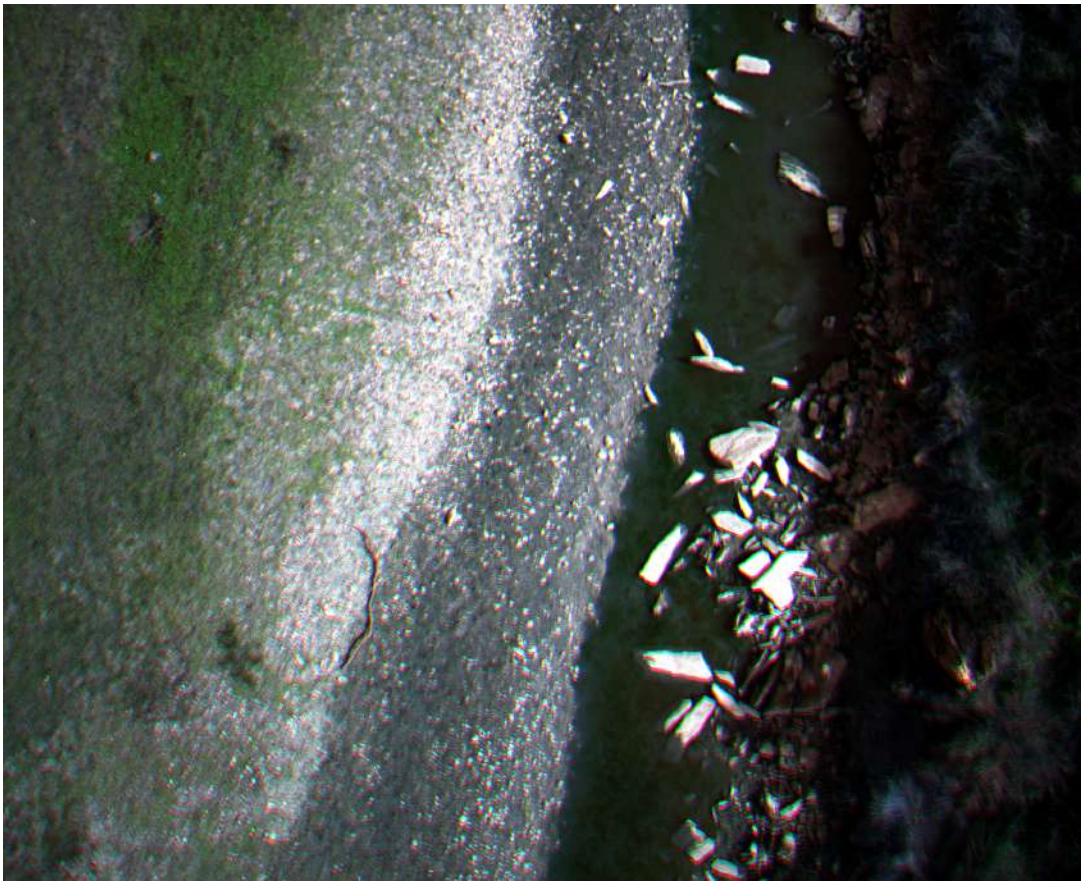


Figure A.1: Image created by stacking bands before alignment.

Figure A.2: Image created by stacking bands after alignment.

A.2 Experiments Results and Data Set Tables

Table A.1: Data set used in experiments.

Exp #	Type	nº train	nº val	nº test
	train	254 total (all 10/15 m)	75 total (all 10/15 m)	36 total (all 10/15 m)
1/11	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	
	train	90 total (all 40 m)	29 total (all 40 m)	16 total (all 40 m)
2/12	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	
	train	17 total (all 70 m)	5(all 70)	5 total (all 70 m)
3/13	val		75(all 10/15)	
	val		29(all 40)	
	val		5(all 70)	
	train	101 total (45 - 10/15 m, 39 - 40m, 17 - 70m)	27 total (11 - 10/15 m, 11 - 40m, 5 - 70 m)	19 total (7 - 10/15 m, 5 - 40 m, 5 - 70 m)
4/14	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	
	train	135 total (45 - 10/15 m, 90 - 40 m)	40 total (11 - 10/15 m, 29 - 40 m)	23 total (7 - 10/15 m 16 - 40 m)
5/15	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	
	train	17 total (all 70 m)	5 total (all 70 m)	5 total (all 70 m)
6/16	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	
	train	101 total (45 - 10/15 m, 39 - 40m, 17 - 70 m)	27 total (11 - 10/15 m, 11 - 40 m, 5 - 70 m)	19 total (7 - 10/15 m, 5 - 40 m, 5 - 70 m)
7/17	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	
	train	344 total (254 - 10/15 m, 90 - 40 m)	104 total (75 - 10/15 m, 29 - 40 m)	52 total (36 - 10/15 m, 16 - 40 m)
8/18	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	
	train	17 total (all 70 m)	5 total (all 70 m)	5 total (all 70 m)
9/19	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	
	train	101 total (45 - 10/15 m, 39 - 40m, 17 - 70 m)	27 total (11 - 10/15 m, 11 - 40 m, 5 - 70 m)	19 total (7 - 10/15 m, 5 - 40 m, 5 - 70 m)
10/20	val		75 total (all 10/15 m)	
	val		29 total (all 40 m)	
	val		5 total (all 70 m)	

Table A.2: HRNet+OCR experiments results on the test data sets.

Exp #	Hight	Pixel Acc	Class IoU	Producer's Acc.	User's Acc.	Pre-train	Pre-train weights	Train time
	10/15m	0.967	[0.957 0.868]	0.905	0.890			
1	40m	0.963	[0.960 0.567]	0.717	0.692	no		33h
	70m	0.961	[0.960 0.302]	0.728	0.458			
	10/15m	0.936	[0.920 0.753]	0.889	0.879			
2	40m	0.988	[0.986 0.839]	0.871	0.851	yes	exp_1	7h
	70m	0.993	[0.992 0.848]	0.881	0.870			
	10/15m	0.825	[0.810 0.302]	0.692	0.898			
3	40m	0.986	[0.985 0.819]	0.868	0.902	yes	exp_2	2h
	70m	0.993	[0.993 0.862]	0.913	0.901			
	10/15m	0.961	[0.949 0.850]	0.929	0.888			
4	40m	0.983	[0.981 0.788]	0.887	0.831	yes	exp_2	13h
	70m	0.984	[0.983 0.696]	0.866	0.751			
	10/15m	0.961	[0.950 0.842]	0.882	0.931			
5	40m	0.979	[0.977 0.736]	0.781	0.892	yes	exp_1	2h
	70m	0.972	[0.970 0.555]	0.780	0.644			
	10/15m	0.839	[0.824 0.342]	0.555	0.900			
6	40m	0.975	[0.973 0.678]	0.848	0.873	yes	exp_5	1h
	70m	0.993	[0.992 0.846]	0.907	0.865			
	10/15m	0.958	[0.946 0.835]	0.877	0.925			
7	40m	0.980	[0.979 0.749]	0.768	0.861	yes	exp_5	1h
	70m	0.987	[0.986 0.720]	0.768	0.850			
	10/15m	0.972	[0.964 0.889]	0.937	0.942			
8	40m	0.983	[0.981 0.783]	0.825	0.905	no		64h
	70m	0.989	[0.988 0.767]	0.798	0.926			
	10/15m	0.917	[0.900 0.661]	0.735	0.941			
9	40m	0.979	[0.977 0.738]	0.825	0.888	yes	exp_8	<1h
	70m	0.992	[0.991 0.823]	0.919	0.870			
	10/15m	0.952	[0.937 0.824]	0.946	0.887			
10	40m	0.984	[0.982 0.800]	0.895	0.831	yes	exp_8	6h
	70m	0.989	[0.988 0.774]	0.864	0.867			

Table A.3: Modified HRNet+OCR experiments results on the test data sets.

Exp #	Hight	Pixel Acc	Class IoU	Producer's Acc.	User's Acc.	Pre-train	Pre-train weights	Train time
	10/15m	0.957	[0.945 0.832]	0.909	0.908			
11	40m	0.964	[0.962 0.601]	0.738	0.765	no		21h
	70m	0.964	[0.963 0.392]	0.526	0.606			
	10/15m	0.913	[0.893 0.671]	0.759	0.854			
12	40m	0.988	[0.986 0.846]	0.921	0.913	yes	exp_11	5h
	70m	0.991	[0.990 0.885]	0.857	0.931			
	10/15m	0.851	[0.834 0.402]	0.426	0.880			
13	40m	0.986	[0.985 0.824]	0.886	0.922	yes	exp_12	1h
	70m	0.992	[0.991 0.896]	0.911	0.899			
	10/15m	0.951	[0.937 0.849]	0.888	0.902			
14	40m	0.981	[0.979 0.758]	0.823	0.906	yes	exp_12	8h
	70m	0.986	[0.985 0.710]	0.759	0.918			
	10/15m	0.960	[0.948 0.843]	0.920	0.910			
15	40m	0.985	[0.984 0.811]	0.873	0.920	yes	exp_11	2h
	70m	0.988	[0.987 0.731]	0.769	0.937			
	10/15m	0.843	[0.829 0.339]	0.343	0.976			
16	40m	0.985	[0.983 0.800]	0.840	0.944	yes	exp_15	1h
	70m	0.991	[0.990 0.878]	0.874	0.915			
	10/15m	0.950	[0.935 0.820]	0.959	0.850			
17	40m	0.987	[0.985 0.830]	0.899	0.916	yes	exp_15	1h
	70m	0.990	[0.989 0.769]	0.799	0.955			
	10/15m	0.955	[0.941 0.831]	0.943	0.875			
18	40m	0.971	[0.969 0.689]	0.872	0.768	no		16h
	70m	0.967	[0.965 0.537]	0.867	0.586			
	10/15m	0.842	[0.827 0.346]	0.355	0.934			
19	40m	0.979	[0.977 0.731]	0.783	0.917	yes	exp_18	<1h
	70m	0.989	[0.988 0.773]	0.854	0.891			
	10/15m	0.869	[0.934 0.803]	0.890	0.892			
20	40m	0.848	[0.974 0.720]	0.836	0.839	yes	exp_18	1h
	70m	0.821	[0.980 0.662]	0.840	0.758			

A.3 Ludvision Dataset Availability Statement

The dataset presented in this study, is available on request from the corresponding author. The dataset is not available in a public repository due to it's large size.

Glossary

Atmospheric Correction	Process of removing the scattering and absorption effects from the atmosphere to obtain the surface reflectance characterizing (surface properties).
Change Detection	Process of identifying differences in the state of an object or phenomenon by observing it at different times.
Computer Vision	A field of AI that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs — and take actions or make recommendations based on that information.
Decision Trees	Non-parametric supervised learning method used for classification and regression. DTs learn from data to approximate with a set of if-then-else decision rules.
Deep Learning	Is a subset of ML, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to "learn" from large amounts of data.
Errors of Commission	Refer to sites that are classified as reference sites but were left out (or omitted) from the correct class in the classified map.
Errors of Omission	refer to reference sites left out (or omitted) from the correct class in the classified map.
Genus	A taxonomic category ranking used in biological classification that is below family and above species. Species exhibiting similar characteristics comprise a genus.

Geometric Correction	Transforms the X and Y dimensions of a remotely sensed image so that original distortions are eliminated or at least minimized and the X and Y dimensions of the output image correspond to a chosen geometric reference system.
Image Classification	The task of associating one (single-label classification) or more (multi-label classification) labels to a given image.
Instance Segmentation	Similar to semantic segmentation task, but it gives a unique label to every instance of a particular object in the image.
Invasive Alien Species	Organisms that are non-native to an ecosystem, and which may cause economic or environmental harm or adversely affect human health. They impact adversely upon biodiversity, including decline or elimination of native species and the disruption of local ecosystems and ecosystem functions.
Intersection over Union (IoU)	Is a metric that allows to determine the extent of overlap between two areas. The value ranges between 0 and 1, and is calculated by dividing the area of overlap by the area of union of the two areas.
Kappa Coefficient	Ranges from -1 to 1 and evaluates how well the classification performed as compared to just randomly assigning values.
Ludwigia peploides	A species natural to South America that invades rivers, ponds, and rice fields. It can grow in deep waters, as a fully or partially submerged plant, and form floating mantles. It prevents the entry of light affecting submerged species and blocking the water lines, affecting navigation, fishing, and recreational use.
Machine Learning	Is the science of getting computers to act without being explicitly programmed, learning from experience to automatically improve computer algorithms.

Neural Networks	Are a subset of ML inspired by the structure of the human brain, and are at the heart of deep learning algorithms. They are designed to recognize patterns through the underlying relationships of features in the training process, molded by trying to reproduce the human brain behavior.
Object Detection	The task of locating the presence of objects in an image, generally with a bounding box, and indicating to which class it belongs.
Overall Accuracy	Tells what proportion of the reference sites were mapped correctly.
Panoptic Segmentation	Classifies all the pixels in the image as belonging to a class label, yet also identify what instance of that class they belong to.
Phenology	The study of periodic events in biological life cycles and how these are influenced by seasonal and inter-annual variations in climate, as well as habitat factors.
Pixel Accuracy	The percentage of correctly classified pixels in the image. The result is the same as the Overall Accuracy.
Producers Accuracy	The map accuracy from the point of view of the map-maker (the producer). This is, how often the real features on the ground are correctly shown on the classified map.
Random Forests	Automated algorithm based on ensemble learning, that builds many decision trees and establishes an output based on the predictions of the decision trees.
Remote Sensing	The process of detecting and monitoring the physical characteristics of an area by measuring its reflected and emitted radiation at a distance.
Semantic Segmentation	The problem of assigning a class label to each pixel, disregarding different instances of the same object.

Spectral Signature	Plot of all the variations of reflectance or emittance of a material as a function of wavelengths. Each substance will have its own unique pattern of spectral lines.
Support Vector Machines	Linear model for classification and regression problems. It can solve linear and non-linear problems and works well for many practical problems.
Users Accuracy	The accuracy from the point of view of a map user. The user's accuracy essentially tells how often the class on the map will be present on the ground.

Declaração de Integridade

Eu, António José Marques Abreu, que abaixo assino, estudante com o número de inscrição 10620 de Engenharia Informática da Faculdade de Engenharia declaro ter desenvolvido o presente trabalho e elaborado o presente texto em total consonância com o **Código de Integridades da Universidade da Beira Interior**.

Mais concretamente afirmo não ter incorrido em qualquer das variedades de Fraude Académica, e que aqui declaro conhecer, que em particular atendi à exigida referenciação de frases, extratos, imagens e outras formas de trabalho intelectual, e assumindo assim na íntegra as responsabilidades da autoria.

Universidade da Beira Interior, Covilhã 26 /06 /2022

Assinado por: António José Marques Abreu
Identificação: BI14042048
Data: 2022-06-26 às 22:45:25