DEPARTMENT OF COMPUTER SCIENCE

MIGUEL ÂNGELO LAGE MADEIRA
BSc in Computer Science

MACHINE LEARNING FOR PRECISION VITICULTURE: FOCUSING ON PLANT HEALTH

MASTER IN COMPUTER SCIENCE AND ENGINEERING

NOVA University Lisbon
July, 2024

DEPARTMENT OF COMPUTER SCIENCE

MACHINE LEARNING FOR PRECISION VITICULTURE: FOCUSING ON PLANT HEALTH

MIGUEL ÂNGELO LAGE MADEIRA

BSc in Computer Science

Adviser: Pedro Albuquerque Santos
Assistant Professor, Lisbon School of Engineering (ISEL)

Co-advisers: Rui Neves Madeira
Assistant Professor, Polytechnic Institute of Setúbal

David Semedo
Assistant Professor, NOVA University Lisbon

Examination Committee

Chair: Prof. Dra Carla Maria Gonçalves Ferreira
Full Professor, FCT-NOVA

Members: Prof. Dr Octavian Adrian Postolache
Full Professor, ISCTE

Prof. Dr. David Fernandes Semedo
Assistant Professor, FCT-NOVA

MASTER IN COMPUTER SCIENCE AND ENGINEERING

NOVA University Lisbon
July, 2024

Machine Learning for precision viticulture: Focusing on Plant Health

This document was created with the (pdf/Xe/Lua)LaTeX processor and the NOVAthesis template (v6.10.18) [Lou21].

Abstract

Precision Agriculture (PA) represents a significant advancement in agricultural practices, focusing on efficiency and sustainability through the integration of cutting-edge technologies. This dissertation introduces a comprehensive software solution, "SmartData", which is integrated into the existing "AgriDash" application to enhance the management and analysis capabilities in agriculture. SmartData leverages satellite imagery to generate intuitive maps showcasing vegetation, water, and humidity indices, crucial for effective crop management. Additionally, it incorporates a machine learning model for the accurate classification of plant diseases from images, facilitating timely and effective disease management.

The system also includes a geolocated disease marker system, providing precise location and detailed information about plant health issues, and integrates meteorological data from local weather stations to aid in decision-making based on weather conditions. These functionalities not only enhance AgriDash but also align with the goals of PA by improving productivity and sustainability.

The dissertation outlines the development of a disease classification system, the creation of advanced visualization tools for interpreting satellite data, and the integration of real-time meteorological data to enhance agricultural decision-making. These innovations have been integrated into the AgriDash application, improving its utility for sustainable farming practices. The relevance and effectiveness of these solutions are further underscored by the acceptance of a related paper for presentation at the 15th International Conference on Ambient Systems, Networks and Technologies (ANT'24).

Keywords: Precision Agriculture, SmartData, AgriDash, Machine Learning, Satellite Imagery, Disease Classification, Data Visualization, Meteorological Data Integration, Sustainable Farming Practices

Resumo

A Agricultura de Precisão (AP) representa um avanço significativo nas práticas agrícolas, com o foco na eficiência e sustentabilidade por meios de integração com tecnologias de ponta. Esta dissertação apresenta uma solução de software abrangente, "SmartData", que está integrada na aplicação "AgriDash"criada para melhorar as capacidades de gestão e análise na agricultura. O SmartData aproveita imagens de satélite para gerar mapas intuitivos que mostram índices de vegetação, água e umidade, cruciais para um tratamento eficaz das culturas. Além disso, incorpora um modelo de aprendizagem automatica para uma classificação precisa de doenças de plantas a partir de imagens, facilitando um tratamento prévio e eficaz nas doenças.

O sistema também inclui um sistema de marcadores de doenças geolocalizados, fornecendo localização precisa e informações detalhadas sobre questões fitossanitárias, e integra dados meteorológicos de estações meteorológicas locais para auxiliar na tomada de decisões com base nas condições climáticas. Estas funcionalidades não só melhoram o AgriDash, mas também se alinham com os objetivos da AP, melhorando a produtividade e a sustentabilidade.

A dissertação descreve o desenvolvimento de um sistema de classificação de doenças, a criação de ferramentas avançadas de visualização para interpretação de dados de satélite e a integração de dados meteorológicos em tempo real para melhorar a tomada de decisões agrícolas. Estas inovações foram integradas na aplicação AgriDash, melhorando a sua utilidade para práticas agrícolas sustentáveis. A relevância e eficácia destas soluções são ainda sublinhadas pela aceitação de um artigo relacionado para apresentação na 15ª Conferência Internacional sobre Sistemas, Redes e Tecnologias Ambientais (ANT'24).

Palavras-chave: Aprendizagem de máquina, Agricultura de Precisão, Classificação de doenças em vinha, Viticultura, Visualização de dados, Interfaces de usuário móvel, Métricas de avaliação

Contents

List of Figures

List of Tables

List of Listings

Acronyms

ANN	Artificial Neural Networks *(p. 12)*
AVIPE	Associação de Viticultores do Concelho de Palmela *(p. 3)*
CT	Computerized Tomography *(p. 20)*
CV	Cross-Validation *(p. 77)*
DBM	Deep Boltzmann Machines *(p. 12)*
DBN	Deep Belief Networks *(p. 12)*
GEE	Google Earth Engine *(p. 30)*
GIS	Geographic Information Systems *(pp. 1, 7, 20)*
GPS	Global Positioning System *(p. 7)*
IoT	Internet of Things *(p. 2)*
ISPA	International Society of Precision Agriculture *(p. 6)*
ML	Machine Learning *(p. 6)*
MMDL	Multimodal Deep Learning *(p. 12)*
MRI	Magnetic Resonance Imaging *(p. 20)*
NDVI	Normalized Difference Vegetation Index *(p. 9)*
PA	Precision Agriculture *(pp. iii, 1, 6, 17, 24)*
RBM	Artificial Neural NetworksRestricted Boltzmann Machine *(p. 12)*
ROI	Return over Investment *(p. 8)*
VAE	Variational Auto-Encoders *(p. 12)*
VRT	Variable-rate technology *(p. 8)*

1

Introduction

The transformation of agriculture from its primordial practices to its current state is a testament to humanity's resilience and innovation. Historically, as societies evolved, so did their agricultural methodologies, moving from merely subsistence agriculture to the development of sophisticated agricultural systems [M97]. The Green Revolution of the 1960s, for example, marked a significant turning point. It introduced modern agricultural practices, hybridized crops, and chemical fertilizers, leading to an increase in agricultural production and addressing food security issues of the time [EG03]. However, with the beginning of the twenty-first century, the challenges facing agriculture have become multifaceted. Climate change, decreasing arable land, increasing population and decreasing water resources have added layers of complexity [God+10]. This required a more refined, data-driven, and technology-driven approach, leading to the emergence of PA. PA, as a concept, emphasizes the use of advanced technologies such as Geographic Information Systems (GIS), drones, sensors, and machine learning to optimize crop yield and resource use, ensuring sustainability and efficiency [ZK12]. As we venture deeper into the era of the Fourth Industrial Revolution, characterized by the fusion of technologies that blur the lines between the physical, digital, and biological spheres [Sch17], agriculture is poised for another monumental shift. This dissertation investigates this transition, highlighting the critical role of Precision Agriculture as a step towards the future. While the challenges may be unprecedented, so are the technological tools at our disposal. The synergy between conventional agricultural wisdom and PA's innovative capacity is the key to a sustainable and food-secure future.

1.1 Motivation

As the world grapples with escalating food demands, environmental degradation, and dwindling resources, the imperative to revolutionize agricultural practices becomes more pronounced. The advent of precision agriculture stands as a testament to the innovative steps taken to address these pressing issues.

In the vanguard of this agricultural renaissance, PA employs cutting-edge technologies to enable meticulous monitoring and management of farming practices. As we approach the year 2050, with an anticipated global population surge of 9.7 billion, the strain on food production systems will intensify [UN]. Traditional farming practices, often reliant on indiscriminate pesticide and fertilizer use, exacerbate environmental damage and can be detrimental to both farmers' health and finances. Conversely, PA champions the optimization of inputs, bolstering yield while simultaneously curtailing environmental impact and expenditure.

At its core, a farm communicates rich data—from soil composition and moisture levels to pest presence—empowering farmers with the information necessary to refine their craft. However, many decisions continue to be influenced by outdated methods that rely heavily on experiential judgment. Such methods, while steeped in tradition, are susceptible to inaccuracies and inefficiencies, potentially leading to over-application of pesticides and subsequent crop and health hazards.

The paradigm shift towards precision agriculture heralds a new era where decision-making is augmented by technology. Integration of Internet of Things (IoT) devices enables seamless data acquisition, reducing the need for continuous on-site personnel. Machine Learning (ML) algorithms transcend traditional data analysis, uncovering patterns imperceptible to the human eye, and facilitating data-driven decision-making.

However, the transformative potential of PA is only realized when the data is translated into a format that is both interpretable and actionable by farmers. This underscores the critical role of data visualization: a farmer must not only access but also understand the data to make informed decisions.

The increasing complexity in vineyard management highlights a significant gap in the technological support available to winegrowers and agricultural consultants. Despite the large amount of data generated in agricultural operations, its effective use in identifying and classifying problems in the field remains a specific challenge. This situation highlights the urgent need to develop tools that not only facilitate more efficient data collection and analysis, but that can also be accessible to users with different levels of technical expertise.

1.2 Context Description

Precision Agriculture (PA), based on the optimal combination of advanced technology and agricultural practices, promotes efficient resource management to maximize production. This approach marks a transition from traditional, intuition-based agricultural practices to modern, data-driven management. The core of precision agriculture is the precise monitoring of each agricultural plot, enabling specific interventions, adjusted to current conditions and future projections, offering a response to the challenges of food security, sustainability and resource preservation [Wig].

The IoT enables full connectivity in agriculture, with sensors in the field that provide real-time data on soil moisture, weather conditions and crop health. At the same time, Machine Learning interprets this data, identifying patterns and predicting trends, which allows farmers to predict and mitigate potential problems [Fou+15]. Furthermore, Blockchain offers an innovative solution for the traceability of agricultural products, reinforcing consumer trust [Tia16].

Portugal, with its rich agricultural heritage characterized by distinctive products such as cork, olive oil and world-renowned wines, is in a unique position to integrate these technological innovations. Despite its strong presence in the export market, it faces challenges such as rural exodus and the aging of the workforce in the agricultural sector, requiring strategic adaptation [Car].

In this context, organizations such as PORVID and Associação de Viticultores do Concelho de Palmela (AVIPE) [Vit84] play a vital role, serving as a bridge between tradition and innovation and promoting the adoption of PA. The collaboration of these associations with our research team is part of a larger project, which has already seen the creation of the AgriDash application as part of this joint effort [Mad+22]. AgriDash, a previously developed tool, is fundamental to this project, not only as a product of previous academic work but also as a critical element for the introduction of new functionalities. This integration will enrich the application with recent advances, further aligning it with the objectives of promoting efficient and sustainable agricultural practices through technology, reflecting the ongoing commitment to innovation in the Portuguese agricultural sector.

1.3 Proposed Solution

In the face of the challenges posed by the vast amount of data in modern agriculture, this dissertation proposes a software solution to simplify and interpret this data, providing farmers and agricultural consultants with the tools necessary for accurate and informed management of their crops.

The solution proposed, SmartData, together with its integration into the existing AgriDash, offers a series of advanced functionalities for agricultural management and analysis. The tool allows farmers to observe and interpret vegetation, water and humidity indices derived from satellite data. Using advanced mathematical expressions applied to specific spectral bands, the platform transforms satellite images into colorful and informative maps. These maps are essential for understanding vegetation health, water presence and soil moisture levels, providing a detailed and up-to-date aerial view that is critical for accurate crop management.

Furthermore, the tool has the ability to classify plant diseases accurately. Through an image-powered classifier model, the platform identifies and classifies diseases, allowing users to take immediate corrective action to mitigate the spread or impact of these problems.

Another crucial aspect of the solution is the creation and visualization of disease markers. This functionality enables detailed monitoring of each disease, providing precise geographic information and additional details through pop-ups. This facilitates the monitoring and efficient management of phytosanitary problems in the field.

Finally, the solution includes analysis of meteorological data, collected from stations closest to the delimited land, which allows users to better prepare and respond to climatic variables that directly affect agriculture.

All these functionalities not only make up the SmartData system but were also incorporated and harmonized with AgriDash, enhancing the impact and usefulness of these applications in the practice of Precision Agriculture. With this set of tools, this dissertation presents a concrete and direct response to contemporary challenges in the agricultural sector, contributing to more productive and sustainable agriculture.

1.4 Main Contributions

The main contributions of this research work are the following:

- **Disease Classification System:** Development and implementation of a machine learning model, capable of accurately classifying plant diseases, through provided images, allowing early identification and effective management of phytosanitary problems.

- **Advanced Visualization Tools:** Implementation of advanced system to visualize satellite data transformed through mathematical expressions, providing farmers with intuitive and interactive maps of vegetation, water and humidity indices.

- **Geolocated Disease Marker System:** Development of a feature that allows the creation and visualization of markers for identified diseases, providing detailed information and precise geographic location through interactive pop-ups.

- **Meteorological Data System:** Incorporation of climate data collected from nearby weather stations, facilitating decision-making based on current and predicted weather conditions.

- **Integration with AgriDash:** Integration of SmartData with the AgriDash application, resulting in a unified solution that not only centralizes all essential functionalities for Precision Agriculture, but also offers a platform that can be scaled and adapted to future needs and emerging technologies.

- **Facilitation of Data-Based Decision Making:** With the tools developed, users are empowered to make more informed choices, thus improving operational efficiency, productivity and sustainability of agricultural practices.

Moreover, a paper focusing on the proposed solution was presented at the 15th International Conference on Ambient Systems, Networks and Technologies (ANT'24)[1] [Mad+24].

[1] https://cs-conferences.acadiau.ca/ant-24/

1.5 Document Structure

The remaining of the document is structured as follows:

- **Chapter 2** - Presents the related work and key concepts for this dissertation, a brief description of previous related articles on Precision Agriculture techniques, Machine Learning, and Efficient Data Visualization.

- **Chapter 3** - Contains an explanation of the overall solution developed for this dissertation. It presents the API construction process, its architecture, how it is organized, use cases studied and implemented in the application, as well as, the standard implementation and description of different CNN models algorithms and data retrieval. It also describes satellite connectivity, data retrieval, and mathematical expression for the visualization of index layers. Lastly, it provides a brief explanation of weather data connection and how to retrieve and handle data efficiently.

- **Chapter 4** - Describes the web prototype initially developed, SmartData, and its integration with the already developed AgriDash application. This chapter goes over the technologies used, the application's overall structure and added functionalities, as well as some mechanics used to fulfil some of the user's necessities.

- **Chapter 5** - This chapter deals with the evaluation performed on the different parts of the solution and describes the way the tests were performed and the feedback from users (farmers and technicians), showing the evaluation results.

- **Chapter 6** - This chapter summarizes the key takeaways from the document and points out areas of the solution that could be enhanced. Additionally, it provides new ideas that might improve the solution.

2

Related Work

This chapter examines several studies, research, and techniques that were relevant during the development of the entire project. Section 2.1 describes the concept of precision farming, techniques, tools as well as means of achieving precise and sustainable farming. The next section (2.2) gives a brief overview of Machine Learning (ML), more specifically, Deep Learning. Section 2.3 talks about Image Fusion, associated strategies, different techniques and main applications. Section 2.4 refers to big data in agriculture, what big data is and challenges related to agriculture big data and ML. Finally, section 2.5 will focus on the effective Visualization of data, key principles and techniques.

2.1 Precision Agriculture

PA is an agricultural concept that is based on measuring, observing and responding to variability within and between fields of crops and aims to specify a decision support system for farm management and to optimize its resources [McB+05]. Thanks to technological advances and new discoveries, this definition of PA has been changing over the years. In 2019 the International Society of Precision Agriculture (ISPA) defined PA as being a "management strategy that gathers, processes and analyzes temporal, spatial and individual data and combines it with other information to support management decisions according to estimated variability for improved resource use efficiency, productivity, quality, profitability and sustainability of agricultural production" [Spr07].

GPS, remote sensing, satellite images, and other technologies allow data collection on soil conditions, climate, and crop growth. This data allows for precise determination of how farmers apply chemicals, fertilizers, and even water. This is essential to reducing costs and minimizing damage to the environment and soil.

PA solutions discussed above can also be sustainable. Unlike PA, Sustainable Agriculture aims to produce food in a way that is environmentally friendly, socially responsible and economically viable. Solutions that are sustainable and precise include:

- **Variable rate technology (VRT)** - Allows farmers to apply inputs such as seed, fertilizer and pesticides only where they are needed.

- **Remote Sensing** - Use of drones, planes and satellites to monitor crop growth and soil health.
- **Precision irrigation** - Using sensors and weather forecast data to optimize water application, reducing water waste and improving crop yields.
- **Agroecological practices** - They combine traditional agricultural knowledge with modern technology to build more resilient and productive agroecosystems.

In short, precision agriculture is a farming management concept that uses technology to optimize crop production. It uses a variety of tools and techniques to store data, analyze it and make informed decisions in order to increase crop productivity, reduce costs and minimize the impact on the environment.

2.1.1 Techniques for Precision Agriculture

As previously stated, precision agriculture uses a variety of techniques to optimize crop production, some of which are GPS-guided machinery, remote sensing, precision irrigation, variable rate technology, agroecological practices, automation and robotics. By using these technologies, farmers can gather and analyze data, and with it make more informed decisions about planting, fertilizing, and harvesting crops. This will lead to a more efficient use of resources and more sustainable farming practices.

2.1.1.1 GPS-Guided Machinery

The combination of Global Positioning System (GPS) and Geographic Information Systems allows the pairing of real-time data collection with accurate position information, leading to a efficient manipulation and analysis of large amounts of geospatial data [gov21].

Multiple GPS-based applications are used for farm planning, field mapping, soil sampling, tractor guidance, crop scouting, variable rate applications, and yield mapping. These allow farmers to plant, cultivate, and harvest with high accuracy. GPS also allows farmers to work during low—or no-visibility field conditions such as rain, dust, fog, and total darkness.

2.1.1.2 Remote Sensing

Remote sensing is the science of obtaining information from an object without physical contact. Monitoring of crop growth, soil health and weather conditions is done using drones, planes or satellites. It can provide farmers with detailed information about the health of crops, which in turn helps to detect and identify various problems such as pests, diseases and nutritional deficiencies at an earlier stage. This helps improve crop yields and reduce costs [WJD20].

2.1.1.3 Precision Irrigation

Irrigation systems are measured by their efficiency, uniformity, and Return over Investment. Precision irrigation is a sustainable agricultural approach that allows the application of water and nutrients to the plant at the right time and place and in small measured doses in order to provide it with optimal growing conditions [NET24]. This approach uses sensors and weather forecast data to optimize water application and reduce water waste.

By using precision irrigation, farmers can ensure that their crops receive the right amount of nutrients at the right time, which can help to improve crop yields and reduce water consumption [Abi+20].

2.1.1.4 Variable Rate Technology (VRT)

Variable-rate technology (VRT) allows farmers to apply inputs such as fertilizer, chemicals, lime, gypsum and irrigation water at different rates across a field. Thus, farmers are able to reduce the amount of inputs they use, which helps to reduce costs and minimize the environmental impact of farming.

VRT can be used to deal with spatial variability between management zones. There are two types of VRT: Map-Based Control is a map with application rates that is produced for the field prior to carrying out the intended operation and Real-time control where decisions about which rates to apply in different locations are made based on information collected during the operation. This requires sensors to detect the necessary information in real-time. It is usually designed for a specific job, such as herbicide application [ML21; Sah+19].

2.1.1.5 Agroecological practices

Agroecological practices combine traditional agricultural knowledge with modern technology to build resilient and productive agroecosystems. Crop rotation, cover cropping, reduction of chemical inputs and use of more efficient irrigation methods are some examples of principles to be adopted within agroecological practices [WCC14].

2.1.1.6 Automation and Robotics

In precision agriculture, automation and robotics have become one of the main frameworks that aim to minimize the environmental impact and, simultaneously, maximize agricultural production. Tasks such as planting, harvesting and plant monitoring are now performed by automated robots. Currently, the main area of application for robots in agriculture is in the harvesting stage. Over the past few decades, a significant amount of research has focused on mobile robot applications for agricultural operations such as planting, inspecting, spraying, and harvesting [MH20].

The objective of agricultural robotics is to help the agricultural sector in increasing its efficiency and profitability of the processes. The shortage of labor, increased consumer demand and high production costs are some of the factors that have accelerated automation in this sector. Incorporating robotics into agriculture improves not only productivity, but also enhances working conditions for both farmers and workers. Intelligent systems are becoming the ideal solution to drive precision agriculture forward, where Artificial Intelligence provides predictive data to help optimize farms and crops.

2.1.2 Tools for Precision Agriculture

Based on the technologies previously discussed, PA relies on a variety of tools to gather and analyze data: GPS, Sensing Devices, VRT controllers, Farm Management Software, and Automation and Robotics.

- **GPS technology**: It is used to guide tractors and other farm machinery, allows high accuracy plantation, cultivation and harvesting. It also enables farmers to create high-resolution maps of their fields, which can be used to identify variations in soil quality and crop growth.
- **Remote Sensing Devices**: It includes drones, planes and satellites which are used to monitor crop growth, soil health and weather conditions. They can gather information about Normalized Difference Vegetation Index (NDVI) [NAS00] that can be used to measure and estimate crop health and yield potential.
- **Sensors**: It includes meteorological sensors, soil moisture sensors and nutrient sensors. They are used to collect data on the environment and crop health.
- **Variable Rate Technology (VRT) controllers**: These devices control the application of seeds, fertilizers and pesticides in real time based on data collected by sensors.
- **Farm Management Software**: This type of software allows farmers to store, analyze and manage data collected by various precision farming tools such as record keeping, crop mapping and yield analysis.
- **Automation and Robotics**: These include autonomous tractors, harvesting robots and field monitoring robots, that can performs planting, harvesting and monitoring.

2.1.3 IoT-based devices

Internet Of Things (IoT) is an interconnection of a large number of devices that are armed with sensors, software and processing capabilities that communicate with each other through the internet, sharing data about themselves, the environment and other devices in their vicinity, without requiring human to human, or even human to computer communication, and ultimately integrating the physical world with the digital one [Yan11].

This technology has been used in many areas, such as health care, logistics, smart-homes and smart-buildings. In agriculture, IoT-based devices are becoming increasingly popular as a way to improve crop production and efficiency. These devices make use of sensors, wireless communication and cloud-based data analytics to collect and analyze data about the environment and crop growth. These devices are embedded systems composed of a communication module, microprocessors, I/O interfaces, and memory. The connection between devices and sensors is made through Wireless Sensor Networks (WSN) which, thanks to its inherent capabilities of self-organization, self-correction, self-configuration and self-diagnosis, has been the target of exploration when trying to achieve a way to produce food with less waste and maximum efficiency [Eli+18].

The various types of existing sensors can be classified according to their location, the type of mechanism (mechanical or not), whether they are optical and according to the type of variables/data to be monitored and measured. These variables can be soil nutrients, temperature, humidity, wind speed, solar radiation or wind speed [SC10]. For a good use of these devices and sensors, they should not require a lot of maintenance on the part of the farmers. This is due to their possibly isolated location and the large number of sensors needed to obtain better results [Eli+18]. Nevertheless, they should also have great memory capability, computational efficiency, power efficiency, portability, coverage, reliability, durability, and low cost.

IoT-based devices in agriculture can provide farmers with valuable data that can be used to make more informed decisions about planting, fertilizing, and harvesting crops. Some applications of IoT in agriculture include: monitoring of crops [Gar+21], tracking and tracing of supply chain and machinery using autonomous vehicles, monitoring and detection of pests and diseases [Ste+20] and smart irrigation [Lio+21]. This will lead to improved crop yields, reduced costs, and to more sustainable farming practices.

In conclusion, IoT-based devices are becoming increasingly popular in agriculture as a way to improve crop production and efficiency. According to the UN Food and Agriculture Organization, with the exponential growth of world population, shrinking agricultural lands, and depletion of finite natural resources there is a urgent need to produce 70% more food in 2050 and IoT technologies can be the stepping stone to archive that goal [Con22].

2.2 Machine Learning

Machine Learning (ML) is a field of research in artificial intelligence that focuses on learning systems and algorithms that learn from data and make predictions or decisions without being explicitly programmed. ML algorithms can be used for a variety of tasks, including classification, regression, clustering, and anomaly detection. Because of its many applications, ML has covered almost all scientific domains, making a significant impact on science and society [RW14].

By exploiting examples or past experiences, ML can generate efficient relationships for data inputs and reconstruct a knowledge schema to analyze large data volumes [Ben+21]. In agriculture, ML can be applied in various ways to improve different aspects such as crop yields, pest and disease detection, and soil analysis. Regarding crop yield, ML models can be trained on recorded weather data, soil conditions, and other factors to predict crop yields. This helps farmers planning for planting and harvesting, and taking more reasonable decisions about fertilization and irrigation. For pest and disease detection, ML models can be used to detect pests and diseases in crops by analyzing images of plants and identifying signs of infestation [DBF22; Ouh+21]. This helps farmers take preventive measures, at a much earlier stage to protect their crops and minimize losses. In soil analysis, ML can be used to analyze soil samples and predict nutrient content, helping farmers to make more informed decisions about fertilization.

Other examples of how ML is being used in agriculture is in livestock monitoring, where ML is used to monitor the health and behavior of livestock, to detect and address any issues early on, and in autonomous farming, models can be used in the development of autonomous farming equipment, such as drones and tractors, which can assist with planting, harvesting, and other tasks. Overall, Machine Learning is becoming more and more prevalent in agriculture and is helping farmers to improve crop yields, reduce crop loss and increase efficiency.

2.2.1 Deep Learning

Deep learning is a specialized set of methods that belong to the Machine Learning family. Contrary to what happens in ML methods, that start by extracting the most relevant features and are later used to create a model that categorizes the objects in the images, in deep learning the relevant features are automatically extracted from images. It then performs an "end-to-end learning", where a network is given raw data and a task to perform, in the case of this dissertation classification of diseases, and it learns how to do this automatically. This makes DL capable of processing unstructured data at maximum capacity, producing high-quality results, and avoiding unnecessary costs.

Another key difference is that deep learning algorithms scale with data, which means that as the amount of data increases, so does accuracy. The increase in data leads the algorithm to train more and thus reducing the error during classification. Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Restricted Boltzmann machine (RBM), and Deep Belief Network (DBN) are among the most known and used methods of DL.

In recent years, Deep Learning has become commonplace in our daily lives, implemented in a wide variety of applications, products and services, such as: Healthcare using image recognition for disease detection; Financial Services to track stocks, detect fraud, and to help manage credit and investment portfolios; and Customer Service through chatbots and voice recognition.

2.2.2 Multimodal Deep Learning

Despite the intense development carried out in terms of unimodal learning, that is, learning through a single modality, it still fails to cover all aspects of human learning. Modality specifies a representation format in which a given type of information is stored. So, the various forms of media that exist (e.g., image, text, video, or audio) relate to modalities, and the representation of these multiple modalities together can be defined as multimodal [GWW19]. The goal of Multimodal Deep Learning (MMDL) is to create models that can process and link information using various modalities [Sum+21].

MMDL models can be used to analyze and learn from a combination of different types of data such as text, images, audio and video. They consist of several branches, each dealing with a different modality of the input data. The different branches are then combined to make the final decision or prediction. This allows MMDL models to make more accurate predictions.

As the amount of multimodal data available continues to increase, MMDL models are becoming increasingly more important. MMDL models can be used in a variety of applications such as image and speech recognition, video recognition, natural language processing and sentiment analysis. Some examples of MMDL application in agriculture are crop forecasting, pest and disease detection, soil analysis, autonomous farming and livestock monitoring.

2.2.3 MMDL Architectures

MMDL can be categorized in 3 main architectures: probabilistic graphical models, Artificial Neural Networks (ANN) and miscellaneous architectures groups.

2.2.3.1 Probabilistic graphical models

Probabilistic graphical models includes Artificial Neural NetworksRestricted Boltzmann Machine (RBM) [Smo86], Deep Belief Networks (DBN) [GT06], Deep Boltzmann Machines (DBM) [SH09], and Variational Auto-Encoders (VAE) [Hou+17].

Restricted Boltzmann Machines are an undirected graphical model where connections are constrained to form a bipartite graph by its neurons. It is a stochastic and generative model with two layers, a visible layer and a hidden layer.

Deep Belief Networks is another probabilistic graphical model that provides a joint probability distribution over labels and observable data [Smo86]. It is a generative graphical model that learns how to extract a deep hierarchical representation of training data. DBN has been widely used in various fields to address representation learning, semantic hashing and data dimension reduction.

Unlike the others probabilistic models, DBM is a completely generative model that can extract features from data with some missing modalities. It can capture multiple layers of complex representations of input data and is suitable for unsupervised learning because it can be trained on unlabeled data [Sum+21].

Lastly, VAE is one of the most famous techniques for complex distributions in unsupervised learning approaches [Hou+17]. It provides a probabilistic way to define observations of the latent space and ensures that this space contains good properties to allow the generative process. It has promising results in producing complex data like faces, handwritten digits, segmentation, feature prediction from static images and speech synthesis.

2.2.3.2 Artificial Neural Network (ANN)

The ANNs were inspired by the computational powers of a single biological neuron. They are at the forefront of creating Artificial Intelligence (AI) and solve complex problems efficiently using their self-learning capabilities. It is used in various activities including: speech recognition, social media filtering, medical diagnosis, machine translation, or video games. CNN, RNN and YOLO are examples of ANN-type architectures and are the most commonly used for video and image description models.

- Convolutional Neural Network (CNN) architecture is a DL algorithm composed of hidden input layers, and output layers to solve complex patterns. Pre-processing required for CNN architecture is much less in contrast to other classification algorithms. CNN was first introduced in 1989 by LeCuN's work [LeC+89]. CNN has various application such as image segmentation, image classification, video processing, object detection, Speech recognition, anomaly detection, medical image analysis, drug discovery, or media recreation.

- The Recurrent Neural Network (RNN) architecture is primarily used to enhance the capabilities of the NN so that they can take fixed-length data inputs to process variable-length sequences. One disadvantage of the architecture is that RNNs can only process one element at a time, and the output of hidden nodes is used as additional input for the next element. This leads to longer processed time and higher resource costs.

- The You Only Look Once (YOLO) architecture was first introduced by J. Redmon et al. whose purpose is efficient detection of objects in real time [JF16]. In a YOLO model, only a single neural network is used for predicting multiple object bounding boxes and for associating class probabilities. Detecting an object is more complex than classification problem because multiple objects and their respective location are detected instead of one at a time.

2.2.3.3 Miscellaneous Architectures

Other miscellaneous architectures to consider include: Support Vector Machine, Generative Adversarial Network and Hidden Markov model. Support Vector Machine (SVM) is traditionally a supervised learning approach of ML and is used to solve big data classification problems to help multi-domain applications [Sut16]. It is characterized as a statistical learning algorithm that solves linear and non-linear classification problems. The algorithm creates a hyperplane or line between data to separate it into classes. The greater the separation between classes, the easier it will be to make a decision.

Generative Adversarial Network (GAN) is a hybrid architecture consisting of two primary components, generator and discriminator networks [Ian+14]. These networks can be in the form of NN such an ANN or CNN. The discriminator network takes the real data and data generated from the generator network and tries to identify whether the data is real or fake. The generator network can produce new output data after training completion, not distinct from actual data.

Hidden Markov Model (HMM) is a statistical model initial used for speech recognition tasks and biological sequences analysis [FI19]. It is used to collect confidential information from sequential patterns. Many ML techniques are focused on the successful implementation of HMMs in speech recognition, computational biology, character recognition, mobile communication techniques, or bioinformatics.

2.2.4 Applications of MMDL

In this section, we will describe some applications of MMDL models in vineyards along with recent research on the subject and the respective articles. These applications include:

- **Crop yield prediction**: Farmers can predict the yield of their crops based on factors such as weather conditions, soil moisture, and fertilization [Gad+20; Jác21].
- **Pest and disease detection**: Identification and detection of pests and diseases helps farmers take preventative measures to protect their crops [Ali+22; Yua+22].
- **Soil analysis**: Analysis of soil samples, and prediction of the nutrient content of the soil, helps farmers to make more informed decisions about fertilization.
- **Livestock monitoring**: Monitoring the health and behavior of livestock, helps farmers to detect and address any issues early on.
- **Autonomous farming**: Used in the development of autonomous farming equipment, such as drones and tractors, which can assist with planting, harvesting, and other tasks [MY21].

2.2.5 Advantages of MMDL in vineyards

Vineyards are an essential part of the global agricultural landscape, producing grapes for wine, juice, and other products. However, managing and maintaining vineyards can be a challenging task as it requires monitoring the health of the vines, the maturation of the grapes and the weather conditions that can affect the growth of the grapes.

As the MMDL can be used to analyze a wide range of data, the data can include images of vines, structured text data from weather forecasts, and audio data from soil sensors.

By analyzing this data, MMDL can help farmers make more informed decisions about planting, fertilizing, and harvesting grapes.

One of the most significant applications of MMDL in vineyards is the use of image analysis to detect diseases and pests. By analyzing vine images, MMDL models are able to accurately identify diseases and pests, such as powdery mildew and black root, which can have a significant impact on vine health and grape quality. By detecting these diseases and pests on an early stage, farmers can take steps to prevent their further spread, which can help improve vine health and grape quality.

Another important application of MMDL in vineyards is the use of text data from weather forecasts to predict weather patterns that may affect grape growth. By analyzing this data, the MMDL can provide farmers with valuable information about the likelihood of frost, heatwaves or other weather conditions that could harm the grapes. This information can help farmers make more informed decisions about when to plant or harvest grapes, which can help improve crop yields and reduce costs.

In addition to image and text data, the MMDL can also be used to analyze audio data from soil sensors to detect changes in soil moisture or nutrient levels. The results can provide farmers with valuable information about soil health, which can help improve grape growth. In conclusion, MMDL models are a powerful tool that can help farmers to improve vineyard management.

By analyzing a wide range of data that includes images, text and audio, this methods can provide farmers with valuable information that can help improve vine health, grape quality and overall vineyard efficiency. As technology continues to advance, the MMDL is likely to become an increasingly important tool for farmers looking to improve the management of their vineyards.

2.3 Image Fusion

We humans capture most of the information around us through vision. It is with this action that we manage to retain and analyze objects, including their color, texture, depth, or dimension. Thanks to the development of photographic cameras and, consequently, images, it is now possible to preserve what surrounds us in real-time. With technological advances, there as been an increase in image resolution, along with the development of different image treatment techniques and filters. This led to the creation of techniques that, through the combination of multiple images of the same scenario, create a single image capable of containing more information than each of the individual images initially taken. The set of these techniques is called Image Fusion.

The purpose of image fusion is to improve the visual quality, resolution, and overall information content of the resulting image. This can be done through several methods, that includes pixel-level merging, feature-level merging, and decision-level merging. Pixel-level fusion involves combining individual pixels from multiple images to create a new image. On the other hand, Feature-level fusion combines the features extracted from multiple images to create a new image. Finally, Decision-level fusion combines the results of multiple classifiers to produce a final decision.

With the increase on availability of powerful computing resources and advances in machine learning and deep learning, image fusion has become increasingly practical and useful in a wide range of applications, such as in medical imaging, remote sensing, surveillance and video compression. Image fusion plays an important role in improving the accuracy and reliability of image analysis and interpretation.

2.3.1 Image Fusion Process

Image Fusion consists of three major steps: Pre-processing, followed by Image Registration and finally Post-processing [Mas+17]. However, the first step will always be image acquisition. These images will be obtained through various imaging modalities, such as drones, satellites, from different points of view or at different times. Following image acquisition is time for pre-process the images. This is done in order to make them more suitable for the image fusion algorithm. The next and most important step is to register the multiple images. This is done by aligning the sequence of images in order to overlay the corresponding resources and details correctly. Image registration can be broken down into different steps:

- **Feature extraction**: This process detects and extracts the features from each image to produce a feature map. These features can be based on intensity, texture, or shape, and will be fused together to create the final image.
- **Decision Labeling or Feature Mapping**: By finding the correspondence between the detected features, similar features are matched to each other. Then a set of decision maps are created with the registered labeled images.
- **Semantic Equivalence**: To check whether different decision or feature maps come from different modalities but contain data with similar meaning. It is mandatory to connect these maps to a common object to later perform fusion. This process is to be ignore if the data source obtained comes from similar kinds of sensors.
- **Image Resampling and Transformation**: Feature maps are transformed to a common scale in order to obtain the final result in a similar representation format.

Then the resulting images are merged together into one containing an improved explanation of the originals images. The final step is post-processing which is linked to quality assessment and evaluation of the final image resulting from the fusion. This evaluation will depend on the type of visualization used during the fusion. For instance, it can be performed through comparisons made with original images or through quantitative measures (e.g., variance or mutual information). This step often requires the involvement of a person or specialist as a way of evaluating the quality and information contained in the final image. Fig. 2.1 shows the necessary steps to perform image fusion.

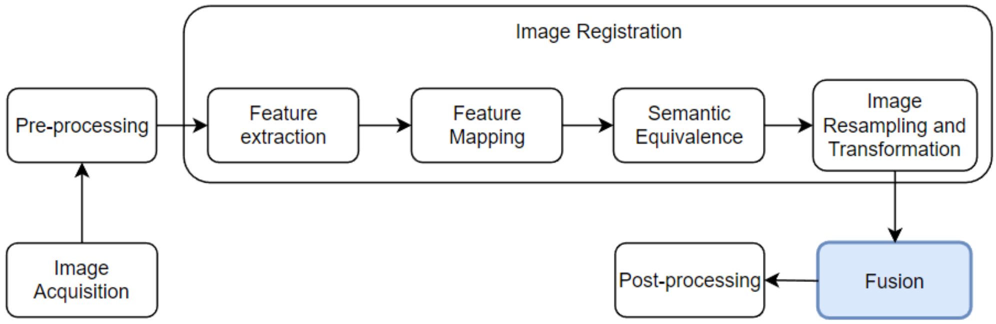


Figure 2.1: Steps for Image Fusion Process

2.3.2 Image Fusion Techniques

There are several image fusion techniques that can be used. These are classified into two groups: Spatial domain and Frequency domain. Masood et al. and Kaur et al. authors describe in detail each existing image fusion techniques, as well as its advantages and disadvantages [KKK21; Mas+17]. There is also a third category of techniques which are deep learning based fusions.

Spatial based techniques are the simplest of the three groups. They deal with the pixel values of the input images where these same values are manipulated to achieve an acceptable result. Examples of Spatial based techniques are: Max–Min, Minimum, Maximum, Simple Average, PCA and guided filtering.

Frequency based techniques decompose the multiscale coefficients of the input images. They can also deal with the spatial distortion of images. Some methods based on the frequency domain are: Laplacian Pyramid, Stationary Wavelet Transform (SWT) and Discrete Transform Fusion Method.

As previously mentioned, Deep learning can extract the most elective features automatically from the data without requiring any human intervention and are able to characterize various complex relationships between targeting data and input data. It was for these reasons that Deep learning based functions have been gaining such popularity in the image fusion area. Several deep learning-based image fusion methods have been presented, showing several advantages for multi-focus image fusion, multi-exposure image fusion, multimodal image fusion, multi-spectral (MS) image fusion, and hyper-spectral (HS) image fusion. Kaur et. al states that multi-view deep learning model was used during the COVID-19 pandemic for validation and testing sets of chest CT images, from data collected from various hospitals in China [KKK21]. Convolutional Neural Network (CNN), Convolutional Sparse Representation (CSR) and Stacked Autoencoder (SAE) are the most commonly used deep learning models in image fusion.

Due to the differences between frequency-based, space-based and deep learning-based techniques, and to achieve the objective of this dissertation, the application of PA techniques and concepts, and due to the nature of the data retrieved from multiple modalities (e.g., multiple sensors, satellite images, and weather stations), the main focus will be on deep learning based fusion techniques. Since it has shown great potential and application in image fusion.

2.3.3 Image Fusion Strategies

In this part, we describe the three main fusion strategies, namely **early fusion, joint fusion and late fusion**. The three frameworks, as well as their extensions, are widely used for deep learning-based fusion [Zha+19]. Figure 2.2 illustrates the three image fusion strategies using deep learing.

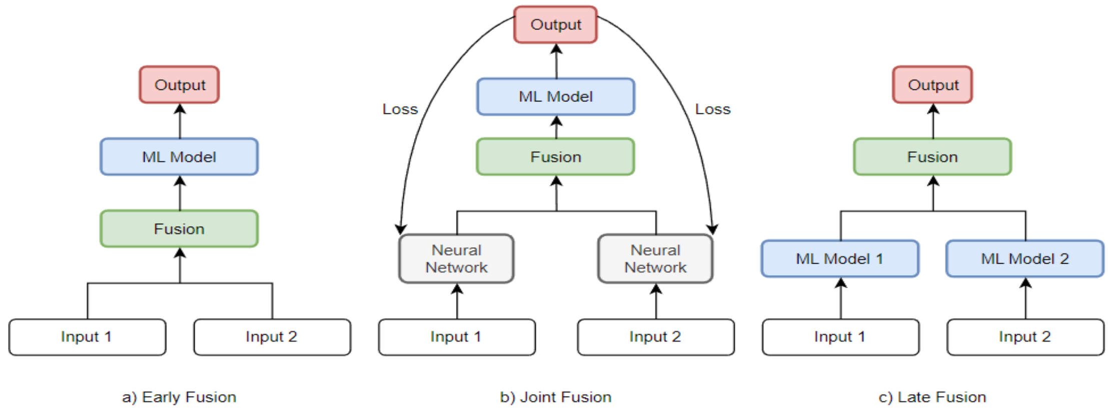


Figure 2.2: Fusion strategies using deep learning

2.3.3.1 Early Fusion

Also known as feature level merging (see Figure 2.2 a)) , refers to the process of merging multiple input modalities into a single feature vector before feeding it to a single machine learning model for training [HPS20]. Input modalities can be joined in a number of different ways, including concatenation, grouping, or applying a gated unit. There are two types of early fusion [HPS20]. Early Fusion Type I, where original features are fused together. Early Fusion Type II, it is the extracted features that are fused together.

This fusion architecture is quite simple. However, it is also more likely to over-fit when data samples are not huge enough. One way to counter this would be to provide larger samples of data to the classification models as a way to get less in-sample error [Zha+19].

2.3.3.2 Joint Fusion

Joint fusion is the process of joining representations of learned features from intermediate layers of neural networks with features from other modalities as input to a final model (see Figure 2.2 b)). Same as Early fusion, there is also Joint fusion type I and II. The main difference, compared to initial merging, is ensuring better feature representations during each training iteration. To ensure this advantage, it is necessary that the loss be propagated back to the feature extracting neural networks during training. This type of fusion is mostly implemented with neural networks due to its ability to propagate loss from the prediction model to the feature extraction model(s).

2.3.3.3 Late Fusion

Also known as as decision-level fusion (see Figure 2.2 c)), because it refers to the process of combining predictions from multiple models to make a final decision. Each modality is trained separately, each with its own models, the final decision later determined by using an aggregation function to combine the predictions from each model. The choice of aggregation function varies depending on the application and input modalities [HPS20].

Compared to the other two approaches, this strategy has the advantage of each network being trained separately, which reduces the difficulty of fitting the model, as well as producing better results. In addition, this model is scalable and flexible, it can be designed according to the requirements and easily designed for multi-input without a large increase in size.

2.3.4 Image Fusion Models

There are different models of image fusion, the main ones being: Single sensor, multi-sensor, multimodal, multi-view, multi-focus, and multi-temporal. Several examples of methods for each of the models have been identified by Kaur et al. [KKK21].

- **Single Sensor**: Several images are fused together to produce a fused image with the best possible information. Human preceptors cannot perceive target objects in lighting and noisy environment which can be highlighted in the final fused image [KKK21]. The limitation of this type is that the resolution of images is dependent on the type of sensors and the conditions in which the system is operated. For instance, digital camera is appropriate for illuminated day-light scenes but is not appropriate for nighttime environments or under poor visibility (fog or rain).

- **Multi Sensors**: As the name implies, it uses integration of images from a number of sensors to form a fused image. It manages to overcome the problems of having only a single sensor. An example of multi-sensor fusion is the use of an infrared camera for low-light environments and the use of a digital camera for day-light views. It is widely used in machine vision, medical imaging, robotics and object detection.

- **Multi-view Fusion**: Images have multiple views at the same time. These approaches are often unable to discard low quality estimations which may result in not obtaining an acceptable level performance. It is also known as Mono-modal fusion.

- **Multi-modal Fusion**: By definition it is the process of integrating multimodal images from one or more imaging modalities to improve the quality of an image. The various models can be multi-spectral, panchromatic, infrared, remote sensing and visible images. Kaur et al. discusses the different Multi-modal techniques explored by several authors [KKK21].

- **Multi-focus Fusion**: It is an efficient method for integrating the information from several images with a similar sight into a wide-ranging image. The compound image is more informative than input images [KKK21]. It provides better visual quality of an image over the other models. By merging and blending the intended details and features from two or more images into a single images ensures that every feature or object in the result image has a proper focus
- **Multi-temporal Fusion**: It captures the same scene at different times. One of the best ways to capture multi-temporal data is through satellite observations. However, it is essential to have constant observations in the long and short term due to changes that occur in the terrain. This type of model is essential for detecting land surface erosion in large geographic areas.

2.3.5 Existing applications

There are several image fusion applications in different fields. This technology is constantly evolving and new applications are being developed as researchers and engineers continue to find new ways to combine and use image data.

Some of these applications are in the field of medicine, where image fusion is used to combine different modalities (CT scan, MRI) to create a more detailed image of the patient [HPS20]; in surveillance, where image fusion used to combine images from multiple cameras and sensors to create a more complete view of the monitored area [HS21]; in computer vision, image fusion is used to combine images from multiple cameras and sensors to create a more accurate and informative view of the environment for object detection and recognition [Zha+19].

In agriculture, image fusion has been used to combine images from multiple sensors, to create a more accurate and informative map of crop growth and health [Li+21]. The area where image fusion has the most impact is in robotics and automation, where it is used to combine images from multiple cameras and sensors to create more accurate and informative views of the environment for the robot to navigate [Zha+17]. Other examples of image fusion application are in Industrial inspection, Remote sensing, GIS and in Augmented Reality.

2.4 Big Data in Agriculture

Big data is a term used to describe the vast amount of data that is generated and collected from various sources. The use of the Internet of Things (IoT) and sensory devices, results in a massive amount of streaming data, i.e., "big data", which can bring new opportunities to monitor agricultural and food processes [Mis+22].

Big Data in agriculture refers to a collection of technologies aimed at addressing the challenges posed by the increasing availability of data. It enables farmers to respond to the demands of the modern data era. By collecting and analyzing a large amount of data, and by incorporating machine learning, farmers are able to make decisions based on more accurate and comprehensive information, addressing problems related to planting, fertilizing, and harvesting.

This large amount of data can come from various sources such as satellite imagery, meteorological data, sensor data from IoT-based devices, and even social media. The use of advanced algorithms and machine learning techniques on this data allows identification of patterns and trends, which helps farmers optimize their agricultural production.

Some examples of big data used in agriculture are: Weather data to predict weather patterns and take necessary actions to protect crops; Satellite imagery that allows farmers to monitor their crops, spot signs of stress, pests or disease early on, enabling them to take action before it is too late; Sensor data from IoT-based devices which helps farmers to monitor soil moisture and nutrient levels, optimizing irrigation and fertilization.

Although big data analysis is leading to advances in various industries, it has not yet been widely applied in agriculture [KKP17]. Nevertheless, the use of big data has the potential to greatly improve the way farmers manage their crops and livestock. As technology continues to advance, more and more farmers are expected to turn to big data to improve the efficiency and sustainability of their operations.

2.4.1 Big Data

Big Data is defined in five dimensions (the five **Vs**) [Cra+22]. The first refers to the enormous **Volume** of data generated, stored and processed. The second is the high **Velocity** of data transmission in interactions and the rates at which data is generated, collected and exchanged. The third refers to the **Variety** of data formats and structures (structured, semi-structured and unstructured) resulting from the heterogeneity of data sources [SOA19]. The fourth dimension is **Veracity**, which refers to the ability to validate the quality of the data used in the analyses. The last dimension refers to the **Value** of the data. This value is obtained by analyzing the data in order to extract hidden patterns, trends and knowledge models through intelligent algorithms and data analysis techniques.

In practice, Big Data analysis tools allow data scientists to discover correlations and patterns by analyzing large sets of data from different sources [Els+18]. In agriculture, big data refers to all the modern technology available combined with data analysis as a foundation for making decisions only based on data [Sar+19].

Big Data has been used to improve various aspects of agriculture, such as knowledge about weather and climate change, land, animal research, crops, soil, weeds, food availability and security, biodiversity, farmer decision making and remote sensing [KKP17]. Another use for big data is in creating platforms that allow supply chain actors to have access to high-quality products and processes, tools to improve yield and forecast demand, while advising and guiding farmers based on the response of their crops to fertilizers, leading to more efficient fertilizer use.

However, big data does not function on its own. It has been used with ML, cloud-based platforms, image processing, NDVI vegetation indices and geographic information systems (GIS) [KKP17]. ML models are mostly used in prediction, grouping, and classification problems, while image processing has been used when the data are extracted from images (cameras and remote sensing) [KKP17]. To achieve the objective of this dissertation, big data tools and ML models will be used, in this case Multimodal Deep Learning models, in conjunction with different types of data taken from sensors (e.g., NDVI, weather stations, soil sensors, and wind sensors).

2.4.2 Challenges in Agricultural Big Data and ML

As mentioned earlier, the use of big data and machine learning (ML) in agriculture has the potential to revolutionize the way farmers manage their crops and livestock. However, there are several challenges that need to be addressed to fully capitalize this potential. The most important ones are: Data Quality and Quantity; Data Privacy and Security; Data Integration; ML Models and Algorithms; Access to Computing Resources and Lack of Expertise.

2.4.2.1 Data Quality and Quantity

Since the agricultural data sets contains various information about soil, climate, seeds, cultivation practices, irrigation facilities, fertilizers, pesticides, weeds, harvesting, and others, collecting high-quality and relevant data becomes a major challenge in agricultural big data [Wee+18]. Due to this multimodal nature of data, produced through IoT and wireless sensor networks, data can be error prone, inaccessible, unusable, incompatible and inconvenient. An example of this is the lack of data interoperability that prevents the integration and unified analysis of data collected by multiple sensors and platforms [Cra+22].

It is important to ensure that sensors are always calibrated, that data is accurate, complete and up-to-date. Baht et al. show that most Big Data systems are better suited for large industrial farms because they have the infrastructure to access data, resources and finance [BH21]. They found only few, if any, examples of small farming operations in the developing world. As mentioned before, thanks to the association with AVIPE, PORVID, and the integration in the European MULTISENS^{2}E project, it is possible to address these problems, making it conceivable to apply and test ML and big data methods in small and medium farms.

2.4.2.2 Data Privacy and Security

However, as the amount of multimodal data increases, concerns about data security and protection arise. The data generated by sensors and satellites is huge and complex, making it difficult to manage intelligent analysis procedures. Ensuring data protection against unauthorized access or breaches is critical to maintaining confidence in agricultural big data and ML. This is particularly important for farmers who may be dealing with confidential information about their crops and livestock.

Although there are some farmers willing to share their data under certain conditions, many still express concerns and even refuse to share data as a way to protect their profits and production techniques [Wee+18]. Therefore, data must be collected consistently and must comply with protocols that can group them into centralized servers. These servers must be protected against cyberattacks while concealing the identity of the farmers.

2.4.2.3 Data Integration

As stated in Data Quality and Quantity (see 2.4.2.1), the fact that there is an increase in large agricultural data, collected from various sources and in different formats, triggers an increased difficulty in data integration and analysis. Cravero et al. identifies different scenarios and challenges where agricultural Big Data is used [Cra+22]. They concluded that the lack of data interoperability, prevents the integration and unified analysis of the data that are collected by various sensors and platforms.

Lassoued et al. analyzed the impact and potential of Big Data in agriculture [Las+21]. They state that there is an enormous difficulty in harmonizing and compiling data from various sources. That may include data coming from satellite imagery, weather forecasts, sensor data and social media. This is due to the fact that there is no standard by which data is captured. The ability to integrate and analyze data from multiple sources remains a major challenge in PA.

2.4.2.4 ML Models and Algorithms

In the wake of data quantity and data integration, it is necessary and important to develop accurate and reliable ML models and algorithms to handle agricultural big data. Selecting the right algorithms and models for the task at hand, as well as dealing with the complexity and variability of agricultural data, is a key step in integrating ML and big data in agriculture.

Studies conducted by White et al. and Bhat et al. indicate that there is a need to develop better representations of crop growth models and the existence of more specific weather forecasts for individual farms and fields [BH21; WTA21]. The complex and volume of the compiled data set studied in this project, makes it challenging to implement procedures of smart analysis. It is expected that scalable and versatile methods can adapt to large amount of information.

2.4.2.5 Access to Computing Resources

As explain in Data Quality and Quantity and Data Integration (see section 2.4.2.1 and 2.4.2.3), not all farms have the same type and amount of funding and resources. Large farms have access to better and larger infrastructures to access data, more resources and financial headroom. On the other hand, running ML models and analyzing big data requires significant computing resources, which can be a challenge for farmers who own small or medium-sized farms that may not or cannot cover the associated expenses or simply do not have access to the necessary hardware and software.

2.4.2.6 Lack of Expertise

More important than resources such as software and hardware, is the training and experience of farmers. Agriculture is a specialized field and the lack of experience in big data and ML can be an obstacle that prevents farmers from fully adopting these technologies.

The implementation of big data in an organization will depend on a clear strategy as well as the existence of trained personnel capable of managing large volumes of data. An example of a problem that exists is that some farmers do not correctly labeling production data on the farm, or not considering all seeding data. Training and talent, more than capital, are fundamental for optimal production in the future [Las+21].

2.5 Efficient data visualization

As we saw in previous sections, agriculture contains vast amounts of data from various modalities. It is necessary for farmers and researchers to understand these data in order to make informed decisions, such as when to apply pesticides and fertilizers, which plants to apply, or which plot of land has the best conditions. All these decisions are part of PA techniques. The best way to help farmers and advisers understand data is to provide an efficient **Data Visualization**.

Data Visualization is an interdisciplinary field that deals with the graphic representation of data [Han23]. Effective data visualization is the process of visualizing data in a way that is easily understandable and interpretable by humans. It is possible to observe patterns and relationships that would often go unnoticed in a more information-based presentation. With the correct presentation of the data, the tasks of the farmers can become much easier if the data is presented in a concise and clear way. The presentation should help in understanding large amounts of information and consequently allow to identify patterns, trends and perceptions that can inform and help in their actions. This is why efficient data visualization is very useful in agriculture.

Standard sketches and diagrams used with Microsoft Excel, Word, PowerPoint are becoming increasingly obsolete. There are innovative techniques present data in more diverse and refined ways with data illustrations, stream graph, heat maps and geographic maps, being that these last ones incredibly useful for farmers [Han23; Kum+19]. Nevertheless, for a good data visualization and understanding of ML and IF results the main focus will be on 2D and 3D visualization techniques.

2.5.1 Data visualization in agriculture

Applying data visualization techniques in agriculture is not an easy process. It requires knowledge and understanding of several key principles that will help in making decisions and subsequently ensuring a good understanding of the data by farmers and advisers. It is also necessary to take into account the environment where each one is when analyzing the data, i.e., farmers spend more time in the fields while advisers spend more time in offices. Farmers just need simpler and quicker information that is straight to the point and takes into account conditions of low visibility (e.g., rain or intense sunlight). Meanwhile, advisers require more detailed information possibly with interactive graphics. The main key principles to be adhered to are the following:

- **Simplicity**: Especially for farmers, keep the view simple and uncluttered, focus on the most important information and avoid extraneous details.
- **Use appropriate visual coding**: Like size, color and position, to represent different dimensions of data. Farmers will analyze the data through an application on their mobile phones when they go to the fields. Such a device is equipped with a small display screen when compared to a computer which leads to an adjustment in the viewing position and reduction in size.
- **Choose the right visualization type**: Probably the most important principle is to select the best visualization type such as bar charts, line charts or heat maps, based on the type and structure of the data.
- **Interactivity**: Such as zooming, panning, and highlighting, allows users to explore the data and gain insights on their own. This can very useful for both farmers and advisers. While it is more beneficial for farmers to provide static, in-the-moment, real-time visualization of data, it can also be efficient to allow them to zoom or highlight some aspects for better visualization of graphs or table values. For advisers, who spend most of their time with access to computers, it would be advantageous to ensure that they can not only zoom and highlight, but also link multiple plots to allow for other types of analyzes and conclusions.
- **Legibility**: Another important principle, without it the representation of data becomes illegible and incomprehensible. Use clear labeling, annotations and color options.
- **Context**: Provide the minimum amount of context and background information needed to help the user understand the data and its implications. For instance, a legend, title and description of the data to be visualized
- **Tailored User Experience**: This principle considers the specific needs and constraints of users, such as farmers and advisers, including their cognitive and perceptual

limitations. Additionally, it addresses the optimization of visualizations for different devices, whether mobile phones or personal computers.

To effectively use data visualization techniques in agriculture, it is critical to follow the principles of effective data visualization described above. Furthermore, it is important to use visualizations in conjunction with data analysis and machine learning to fully understand the data and draw accurate conclusions.

By following these principles, it is possible to use data visualization to powerfully communicate complex information and support farmers in making decisions. Examples of data visualization applications in agriculture include: Crop monitoring to visualize crop growth and health, and to monitor the progress of crops and identify areas of concern; Visualize weather data to understand its impact on crops and plan for possible weather events; Visualization of soil data (nutrient levels, pH, moisture) to understand soil health and make informed fertilization and irrigation decisions; View market data to understand trends and make informed decisions about sales and purchases; Monitor environmental data, including air and water quality, to monitor the impact of agriculture on the environment and make informed decisions about sustainability.

2.5.2 On the field vs At home

In the field, farmers have a lot of responsibilities, which makes it very difficult for them to keep track of everything that is going on. This means that sometimes things can go unnoticed. For example, the farmer may forget to apply pesticide to some plants, resulting in monetary losses to the farm; or they may not apply the correct pesticide in the exact and recommended amounts. These are some examples where the use of notifications to advise and warn the farmer becomes very practical and handy.

Even so, care must be taken not to overload the farmer with information and notifications. The farmer must have access to quick, clear and concise information so that he can act at any given moment and as quickly as possible, depending on the type of information he receives. Therefore, notifications need to be efficient in the way they communicate the problem so that they can be quickly responded to.

An example of simple understanding would be in the case of the appearance of disease such as Mildew, which is vine disease that needs specific conditions to grow according to the rule of three: 10º C of temperature, above 10mm of precipitation and buds above 10cm. In this case, as soon as the conditions are favorable for the disease the farmer would receive a notification with the warning "conditions favorable for the growth of Mildew". By clicking on it will lead to a new screen with the existing solutions to treat the disease, as well as ways to prevent it in the future.

Another example of great data visualization for farmers is the use of bar charts instead of pie charts. Humans can more easily process line length differences than surface areas [Han23], which makes it easier to observe and draw conclusions from a line length comparison than a surface area comparison when the farmer is in the field.

At home or in the office, the working environment is different. There is less noise, you can concentrate more on the work at hand, and you can analyze data, graphs and patterns more easily. Here it is possible to make use of more complex and comprehensive data visualization techniques, such as scatter plots (3D), networks, or flow charts.

2.5.3 Existing Applications

There are many existing applications for data visualization in agriculture. Most are designed to help farmers make more informed decisions and improve their operations by providing real-time insights into crop growth and soil health, as well as production and financial performance. Some popular applications include:

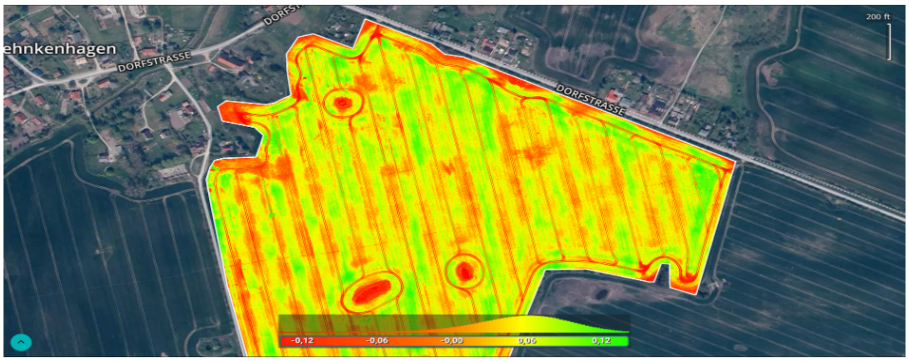


Figure 2.3: PrecisionHawk paired with Pix4D for vegetation indices visualization [ED10]

precisionHawk is a commercial drone and data company, founded in 2010, focused heavily on developing software for aerial data analysis and drone safety systems [ED10]. It has developed software that can perform complete aerial mapping to support agronomy, track growth trends, count and size plants, generate prescription maps, identify early indicators of plant stress, and measure zonal farm efficiency. This allows farmers to see how crops are performing in real-time and make informed decisions.

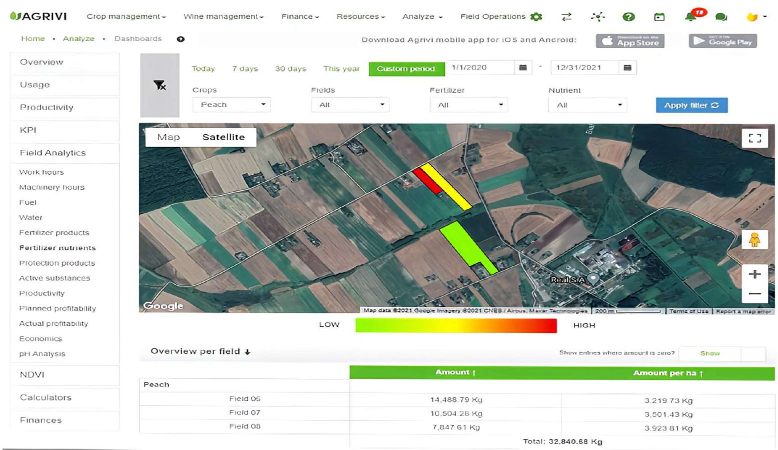


Figure 2.4: Agrivi farm managing software [Agr13]

Agrivi provides real-time insights into crop growth, weather conditions, and soil health [Agr13]. It has some additional features such as easy-to-use and intelligent crop planning with an overview of the historical crop rotation, helps farmers choose the best crops per field for their season, using powerful analytical methods that can analyze the performance of fields, workers and agronomic practices to identify possible improvements for cost optimization and higher yields. It also allows to make timely and informed decisions with real-time satellite field images, weather reports, risk alarms and crop progress monitoring.

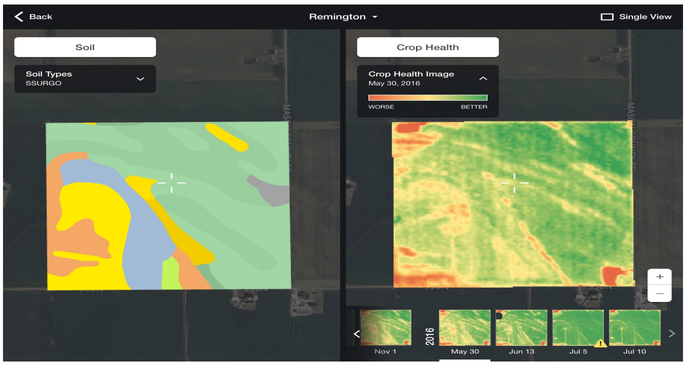


Figure 2.5: Example of visualization of soil types and crop health data [Bus11]

FarmLogs is a powerful, easy-to-use, mobile and desktop application that provides farm management software [Bus11]. It helps farmers manage digital farm records, monitor field and crop conditions, calculate cost of production to market gain with confidence, and analyze their farms' financial performance to the field level. Another key feature is sending alerts in the presence of adverse weather conditions or yield threats.

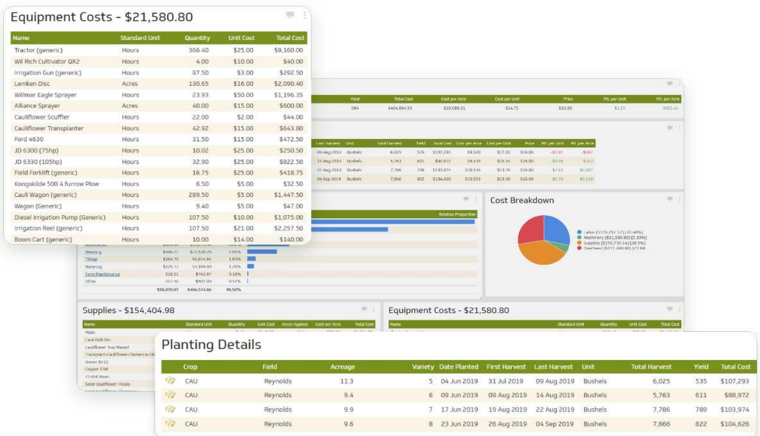


Figure 2.6: Example of using dashboard and reports for farm managing software [Fro09]

The **AgSquared** application gives the entire farm team, field managers, agronomists, tractor drivers, and crew leaders access to the information they need to make informed decisions [Fro09]. It has a mobile field log to simplify field operations, so everyone knows what they need to do, and everything is recorded for future studies. It uses powerful visual analytics such as reports and dashboards to help understand the business, make informed decisions, and document compliance.

3

Design and Implementation

This chapter delves into the intricate details of the design and implementation of the technological solution central to this dissertation. It begins with a comprehensive exposition of the solution architecture, followed by Flask API[1], detailing its role as the backbone for the solution architecture. The connectivity between two different environments is examined in detail, setting the stage for a discussion of the use cases that the system addresses. Particular attention is paid to the CNN model, which forms the essence of the data set retrieval and pre-processing steps. Furthermore, the integration of satellite and other geospatial data through the API connection (GEE) is explored, highlighting the visualization layers and their significance to agriculture.

3.1 Solution Architecture

When creating a powerful and flexible application, architecture forms the backbone, creating a harmony between user experience and core functionality. The developed Flask application adopts a bifurcated design, Fig. 3.1, which encompasses both the front and backend components, with a unique emphasis on the backend architecture due to the distinct requirements of the system.

This architecture, has been meticulously designed to meet the unique challenges and requirements of the intended objectives. Each choice, from Python to HTML, was made after careful deliberation, ensuring that this application is not only functional but also scalable and future-proof.

The Backend of the application serves as the bridge where raw data is transformed into valuable insights. Due to the intricate nature of the task at hand, particularly the integration of TensorFlow for deep learning and Google Earth Engine (GEE) for satellite data retrieval, the backend was segmented into two separate working environments. These distinct environments were required by the incompatibilities between TensorFlow and GEE in terms of dependencies and versions of the programming language. The main Backend environment was built with the purpose to interact with GEE and handling the

[1] https://flask.palletsprojects.com/en/3.0.x/

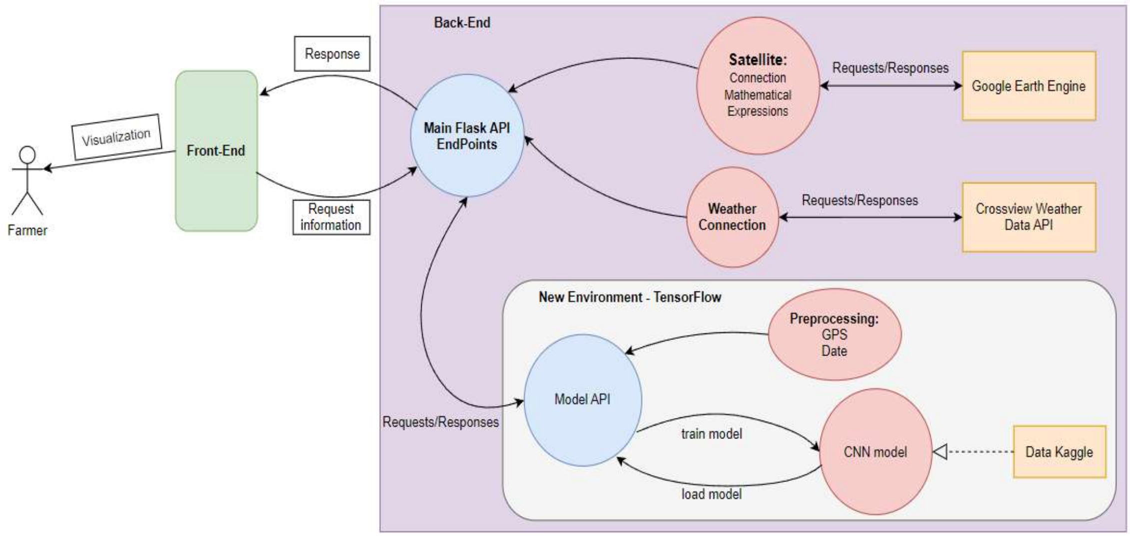


Figure 3.1: Framework of proposed solution

bulk of application logic. Being constructed in Python, it took full advantage of the rich ecosystem of Python, known for its data processing capabilities and its compatibility with numerous satellite data libraries. On the other hand, the microserver environment was exclusively designed with a specialized role, for running and training CNN models. Given the sheer computational demands and unique dependencies of CNNs, isolating this function ensured that the training process remained unhindered by other back-end activities.

The use of Python as the main development language is mainly due to the vast versatility and integration of this language. Its extensive libraries and frameworks make it a one-stop language for deep learning and back-end development on the Web, thus reducing potential points of friction during the development process.

The frontend serves as the user's window into the rich data insights derived from the back-end. Designed using HTML and JavaScript, its main objective is to provide an intuitive and seamless user experience. The choice of these programming languages was seriously considered and driven by some crucial considerations.

- **Simplicity and Universality** - HTML is the standard markup language for documents designed to be displayed in a web browser. Combined with JavaScript, it allows dynamic content rendering without the need for additional tools or frameworks.

- **Map Visualization with Mapbox** - One of the standout features of this application is its map visualization capabilities. Mapbox was the chosen platform for this purpose due to its user-friendly interface and vast repository of pre-built code examples. The natural synergy of JavaScript with Mapbox made it the preferred choice over other languages [Map10].

Although there are other more modern frameworks like React that offer advanced features, the choice to stick with native HTML and JavaScript was strategic. React, while powerful, introduces an additional layer of complexity. Given one of the primary focus, visualization of satellite data using Mapbox, the simplicity and straightforwardness of HTML and JavaScript proved to be more efficient, eliminating potential over-engineering.

3.2 Flask API

In the modern web development scenario, the choice of framework plays a pivotal role in determining the success and efficiency of a project. The Flask API is a testament to this principle. As a web microstructure developed in Python, Flask is known for its simplicity, flexibility, and elegance. Basically, Flask gives the freedom to build applications without the restrictions often found in larger frameworks. But beyond its ease of use, Flask big advantage is that its features provide seamless integration with cutting-edge technologies.

The decision to opt for the Flask API comes from its ability to seamlessly integrate with Convolutional Neural Network (CNN) models. CNNs, a class of deep learning models, have become fundamental in a multitude of applications ranging from image recognition to natural language processing. Once Python has established itself as the language for deep learning – courtesy of powerful libraries like TensorFlow and PyTorch – integration between CNN models and a Flask API becomes almost natural. This synergy enables rapid development and deployment of scalable web applications that can harness the power of real-time CNNs.

Additionally, by choosing Flask, it is possible to stay within the Python ecosystem, ensuring smooth data pipelines, easier debugging, and reduced context switching. This continuity is crucial when dealing with the complexities and dependencies of CNN models, where even small discrepancies can lead to significant performance issues. Thus, Flask serves as a bridge, connecting the world of web applications with the advanced computational resources of deep learning models, all under the umbrella of the Python programming language.

3.3 Connectivity between Environments

In the context of the application's architecture, connectivity plays a pivotal role. The system was designed as a centralized distributed system, where the primary Flask API serves as the main hub, consolidating various functionalities, and acting as the primary interface for external communications. By design, the main API, is the epicenter of the system. It holds all API endpoints, effectively becoming the gateway for all incoming and outgoing data streams. This centralization ensures streamlined management, a unified point of maintenance, and simplified scalability.

One of the more subtle facets of connectivity in the developed system, is the relationship between Flask's core API and dedicated CNN microservers. Every request made from the user interface, or frontend, is first routed through the main Flask API. This layered approach not only ensures a controlled flow of information, but also abstracts the complexities of CNN processes from the end user.

Upon receiving a request that requires CNN processing, the main API communicates with CNN microservers through a predefined set of endpoints. In essence, the microserver itself acts as an independent API, which the main Flask system interacts with. This interaction is facilitated through specific CNN microserver endpoints. So, from an architectural perspective, the main Flask API and the CNN microserver form a cohesive network of interconnected services, communicating seamlessly through endpoints.

By isolating CNN operations on a microserver, updates or modifications to deep learning models can be carried out without affecting the main API, thus promoting modularity within the system. Another advantage would be the distribution of load between the main server and the CNN microserver, thus ensuring that neither system is overloaded, especially during high traffic scenarios. And finally, it guarantees scalability; in the future, if there is a need to introduce servers or more specialized functionalities, these can be easily integrated into the existing system, connecting through the main server.

3.4 Use Cases

By definition, a use case is a list of actions or event steps typically defining the interactions between a role (often referred to as an "actor") and a system, all directed towards accomplishing a particular objective. This section delves into the exploration of various interactions that actors can execute using the developed API. This is depicted through a clear, comprehensive Use Case diagram, figure 3.2, aligning with the standards of Universal Modeling Language (UML).

The illustrative image below showcases the spectrum of operations and interactions possible within the API ecosystem. Each use case within the diagram represents a distinct operation or functionality available to the actors, providing an immediate visual grasp of the API's capabilities.

This image reveals the extensive range of services provided by the API. With actors such as Winegrowers and Technicians in focus, the diagram provides a visual blueprint of permissible engagements within the API, fostering optimized engagement. Highlighting distinct roles and their respective permitted actions, the diagram offers clarity on role-specific constraints and opportunities within the API domain. For third-party entities seeking integration, this diagram serves as a clear roadmap, simplifying integration efforts and enhancing collaboration between systems. With both visual and textual breakdowns of the explored use cases, this section acts as a comprehensive guide, empowering users to utilize the API effectively, ensuring smooth operation and success of integrated systems.

Additionally, this section will explore three key scenarios showcasing the practical application of the API: Disease Classification and Treatment, Weather Data Visualization, and Satellite Data Visualization. Through these scenarios, users can grasp not only the

Figure 3.2: Use Case Diagram

technical capabilities of the system but also its tangible impact on everyday viticulture operations.

3.4.1 Disease Classification

Harnessing the power of deep learning, this scenario elucidates how the system identifies specific diseases that plague the vineyard and offers insights into potential treatment solutions.

For disease classification, users initiate the process by uploading an image of the leaf in question. This image is prominently displayed on one side of the interface. After pressing the button "Classify image" the system will send a request to the ModelAPI system (as depicted in Figure 3.1) to classify the provided image using the already loaded classification model. Once processed, the classification outcome is presented in a textual format, detailing both the potential disease label and its associated confidence percentage. Complementing this, an illustrative image, representative of the identified disease, appears on the opposite side. This juxtaposition offers users, be they farmers or technicians, a comparative perspective: farmers can match their sample against the reference image, while technicians gain insights into the model's accuracy. Such a design not only aids in instant verification but also assists technicians in continually refining the model, ensuring it remains robust and avoids overfitting to the training data.

When a given image has metadata, the Preprocessing phase show in Figure 3.1 is used in conjunction with the image classification process to build a marker containing comprehensive information on the same image. The image itself, the GPS locations, the date the picture was taken, and the disease label to which the classifier assigned are all included in this information. Afterwards, this marker will be shown in the window for visualizing satellite data. Image metadata is text information pertaining to an image file that is stored in a file that is not typically visible to the end-user[2]. This text information includes details relevant to the image itself and to its production. Image metadata is often divided into three main categories: Technical metadata, Descriptive metadata, and Administrative metadata. The main difference of the three is that Technical metadata is mostly created automatically by the device or software that creates the image while the others are added manually using special software. For reference, a digital camera or smartphone will typically produce metadata regarding the camera and the settings used to take the picture. The most crucial information is the time and date of the picture's creation, as well as the its GPS location. Therefore, users are advised to turn on the GPS option when taking a photo to ensure that this information is always available.

3.4.2 Weather Data Visualization

Viticulture is heavily influenced by weather patterns. This scenario illustrates how the API collects and presents important weather data, providing stakeholders with timely information essential for decision-making. Before exploring the data visualization, users need to specify a region or area of interest. Using this information, the API calculates the coordinates of the designated area's central point. By combining reverse geocoding with MapBox capabilities, it determines the nearest city or district to the specified region. The name of the identified city is crucial for accessing meteorological data from the VisualCrossing API.

The visual display shows various weather metrics, such as maximum, average, and minimum temperatures, precipitation levels, wind speeds, and humidity levels. Additionally, the system provides a forecast for the next 15 days and historical data for the past 15

[2] https://www.techtarget.com/whatis/definition/image-metadata

days, offering insights into future and past weather conditions.

3.4.3 Satellite Data Visualization

As with weather data visualization, the initial precondition requires the user to mark a specific plot or parcel of land. When these spatial parameters are set, users are presented with five distinct satellite data visualization alternatives. This information is sourced from the Google Earth Engine (GEE), which specifically uses data from the Copernicus Sentinel-2 satellite mission. These satellite visual layers are constructed using mathematical formulations applied across various spectral bands, which include RGB, near-infrared (NIR) and multiple red spectra.

Central to the implementation was the emphasis on user experience. Consequently, the layers were devised using a tile-based representation rather than static imagery. In the realm of geospatial visualization, tiles serve as modular map units, their resolution governed by the formula 2^{zoom}. Hence, with increasing zoom levels, each tile boasts a higher pixel count, yielding enhanced visual clarity. This dynamic resolution adjustment is designed to harness the full potential of the Google Earth Engine's rendering capabilities, ensuring optimal visualization at different zoom levels.

Moreover, the API is capable of handling multiple drawn polygons (parcels of lands), visualizing different satellite layers within each of those polygons, and also providing opacity modulation feature for these same satellite layers, adds an additional dimension to data interpretation. This visualization flexibility allows users to better understand the underlying satellite data.

3.5 CNN model

The evolution of modern agriculture has been profoundly influenced by advances in technology, particularly in the realm of artificial intelligence and computer vision. Among the various of techniques within computer vision, Convolutional Neural Networks (CNNs) have emerged as the gold standard, demonstrating unparalleled prowess in image-based tasks. These deep learning models have the unique capability of extracting intricate patterns and features from images, making them an apt choice for tasks that require high precision and reliability.

In the context of this project, the value of CNNs plays a particularly important role. Vineyards, as serene as they appear, are hotbeds for various diseases, many of which manifest visually on grapevines, leaves, and fruits. Early detection of such diseases can be the difference between a fruitful harvest and devastating losses. Traditionally, detecting these ailments relied heavily on human expertise, which, although valuable, is prone to subjectivity and inconsistencies. Enter CNNs, which offer a standardized, rapid, and highly accurate means of diagnosing vineyard diseases based on images collected from the field.

The objective of our CNN implementation is clear: to develop a model capable of classifying diseases in vineyards with high accuracy. By training our model on a rich dataset of vineyard images, we aim to capture the subtle nuances and manifestations of various diseases. Once trained, this model becomes part of our Flask API system. Users can effortlessly upload images captured from their vineyards via the frontend interface, and our system, powered by the CNN, quickly analyzes the images and provides insights into potential disease presence and type.

In essence, by integrating CNNs into our system, we are not just leveraging a technological tool; we are building a bridge between traditional viticulture and modern agritech. This integration promises more sustainable, efficient, and prosperous vineyards, safeguarding both our cherished wine traditions and the economic viability of vineyards.

3.5.1 Dataset retrieval

The foundation of any successful machine learning or deep learning model lies in its dataset. The quality, diversity, and volume of this data can dictate the efficacy of the trained model. For our CNN's purpose of classifying diseases in vineyards, a specialized dataset was indispensable.

Our primary dataset was sourced from Kaggle, specifically from the Augmented Grape Disease Detection Dataset[3]. This data set is a comprehensive collection explicitly curated for the task at hand. It boasts a plethora of images, capturing the various diseases that afflict grapevines, ensuring a diverse and representative sample suitable for our needs. It is a balanced dataset, has 12,000 images evenly spread across four classes Black Rot, ESCA, Leaf Blight, and Healthy giving each class 3000 images, as seen in below in Figure 3.3.

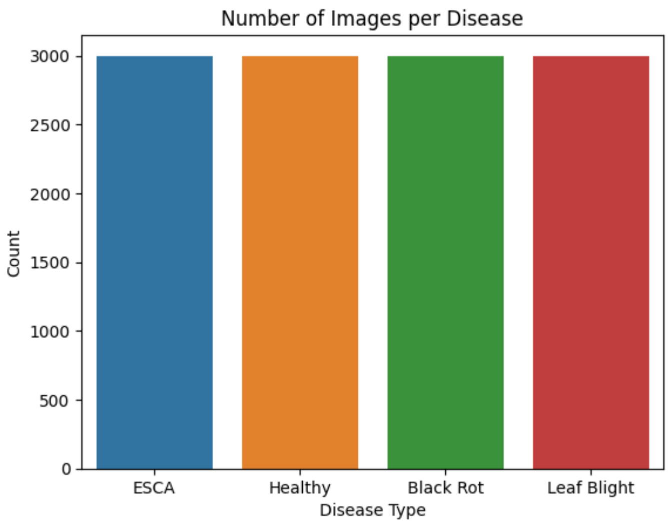


Figure 3.3: Class distribution within retrieved dataset.

[3] https://www.kaggle.com/datasets/rm1000/augmented-grape-disease-detection-dataset/data

Kaggle, as a platform, has become a repository of high-quality datasets, and our chosen dataset is testament to this quality. Using this well-curated and augmented dataset, we not only enhance the potential of our CNN, but also ensure its applicability and reliability in real-world scenarios, perfectly aligning with our objective of efficient and accurate vineyard disease classification.

3.5.2 Data preprocessing

Data preprocessing is the next essential step in the workflow of machine learning and deep learning models. This stage aids in enhancing the quality of data, ensuring its readiness for training, and maximizing the performance of the chosen algorithm. In the context of our vineyard disease classification task, given that we are working with an augmented dataset, understanding the origin and nature of the preprocessing becomes vital.

The Augmented Grape Disease Detection Dataset utilized in this dissertation was derived from a primary dataset called Grape_disease[4]. The primary dataset offered a set of unaltered grapevine images that served as the base upon which the augmented dataset was built. Initially, it contained only 7,222 samples equally distributed in the four classes (Healthy, ESCA, Leaf Blight, and Black Rot). The image below represents the distribution of data across classes; it should be noted that this data set was initially not balanced with the ESCA label with the largest number of samples.

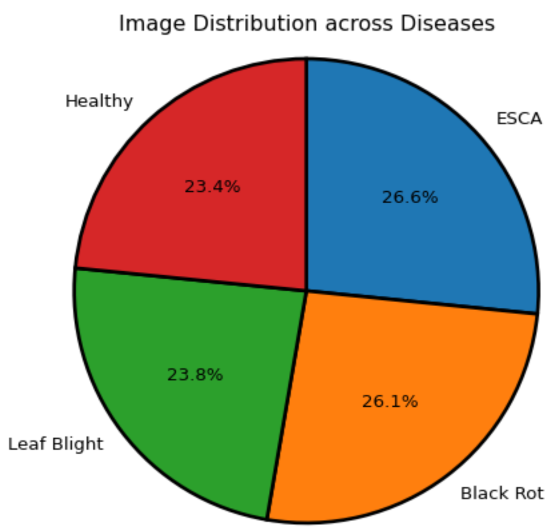


Figure 3.4: Class distribution of initial dataset [Man23].

Augmentation techniques were applied to the primary dataset to increase its size, diversity, and robustness [Man23]. ImageDataGenerator was used to perform a variety of augmentations techniques on the input images. Such techniques included rotations where images can be rotated by any angle between 0 and 360 degrees, zooming, determining the range by which the images can be randomly zoomed in or out; in this case, images can be

[4] https://www.kaggle.com/datasets/pushpalama/grape-disease/data

zoomed in or out by up to 33%, horizontal flip and vertical flip were applied, which helps introduce more variation into the training data, the brightness range was set between 0.5 and 1.5 meaning that the brightness of images was multiplied by a factor between 0.5 and 1.5, and finally the shear range value of 7.5, this parameter specifies the range of shearing angles to be applied to the image; in other words, a transformation that tilts the shape of the image. By introducing these variations, is possible to emulate different scenarios and angles that a grapevine might be seen in real-world conditions. This ensures that the model is trained on a diverse set of data and reduces the risk of overfitting to a specific pattern or style.

After augmentation, certain preprocessing steps were necessary to ensure the dataset's consistency and compatibility with our CNN models. These steps include missing value detection, presence of outliers, and class distribution. Before separating the dataset into training, test and validation set, the number of missing values present in dataset is detected, if any are found the corresponding lines are removed. It is better to remove if there are few than trying to fill them since it can lead to future errors and bad classifications if your not certain of the correct value to fill. After that comes outliers detection, i.e., an outlier is a single data point that goes far outside the average value of a group of statistics, can appear with less or high frequency. In this case, outliers are values that have frequency of less than 1000 in the column *Label* and a frequency higher than 1 in column *Filepath* because images with the same name are not allowed. The code snippet below showcases the steps for detecting missing values and outliers:

```
# Check missing values
missing_values = image_df.isnull().sum()

# Check if any value is greater than 0
if (missing_values > 0).any():
    print("There are missing values in the dataset.")
    # Handling missing values
    image_df.dropna(inplace=True, ignore_index=True)
else:
    print("No missing values found in the dataset.")

#Identifying outliers with very high or low frequencies
#Value with frequency less than 1000 low disease examples
outliers_Label = image_df['Label'].value_counts()
                [image_df['Label'].value_counts() < 1000]

#Value with frequency higher than 1
outliers_Filepath = image_df['Filepath'].value_counts()
```

```
                    [image_df['Filepath'].value_counts() > 1]
```

Listing 3.1: Data preprocessing

Following the preprocessing, the dataset was split into three subsets: 64% for training, 16% for validation, and 20% for testing. This split is a commonly accepted practice and is important for training our model effectively. The training set is large enough for the model to learn the necessary features, while the validation and test sets are sufficient to check the model's performance on data it hasn't seen before.

The first step involves using the *train_test_split* function from the *sklearn.model_selection* module to separate out 80% of the data for training purposes and the remaining 20% for testing. The shuffle parameter is set to True, indicating that the dataset will be randomly shuffled before splitting, to avoid any underlying order in the dataset that could bias the model. The *random_state* is set to 1 to ensure reproducibility of the split.

Within the training data, a further split was made to generate a validation set. This is supported by the *validation_split* parameter set to 0.2 in the ImageDataGenerator configuration, which reserves 20% of the training data for validation purposes.

The *create_generator* function created defines the generators for the training, validation, and test sets using Keras's ImageDataGenerator, which is a useful tool to load images, apply preprocessing functions, and augment the data on-the-fly during training. The *flow_from_dataframe* method of the ImageDataGenerator is used to create iterators that will feed the data into the neural network in batches. The parameters like *target_size*, *color_mode*, *class_mode*, and *batch_size* are appropriately configured to match the needs of the model and the nature of the dataset.

3.5.3 Models

In the landscape of deep learning, especially in the domain of image classification, several architectures have gained prominence due to their exceptional performance on various benchmarks. However, adopting a one-size-fits-all approach can be limiting. Given the unique requirements and challenges presented by viticulture imagery, this work ventured beyond merely adopting existing architectures. Instead, it involved the fine-tuning of established models and the creation of a novel CNN tailored to the task. Three methods were meticulously implemented and modified to meet the specific classification demands of vineyard diseases: a customized CNN model, ResNet50 and MobileNetV2.

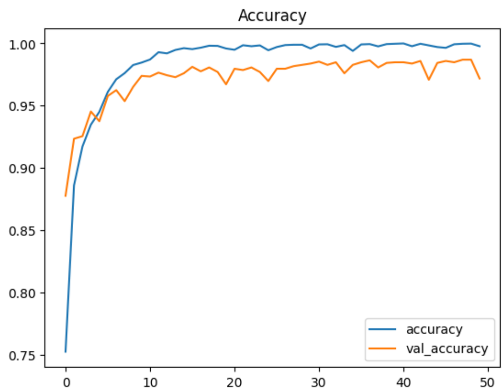
3.5.3.1 CNN created

The custom CNN model was built with the specific goal of accurately identifying and classifying diseases from vineyard images. It was created using TensorFlow and Keras without any pre-existing models to start from. The model has layers that work together to understand the pictures. The first layer, which has 32 filters, looks for simple shapes and edges in the images. As the model processes the data, it uses pooling layers to shrink the image size, which helps it to run faster and focus on the important parts of the image.

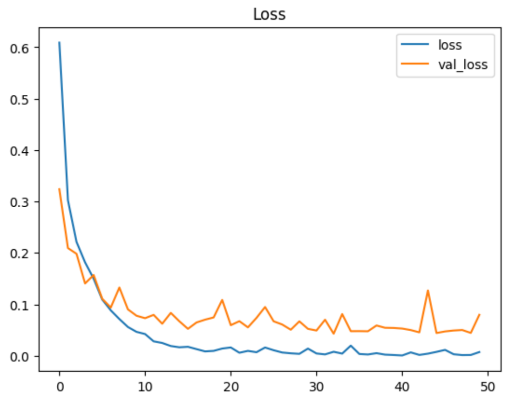
More layers with more filters look for detailed patterns. These layers are crucial to find the specific signs of diseases in the images. The inclusion of max-pooling layers helps reduce the dimensionality of the data, speeding up the processing and focusing the model on important features. To prevent the model from memorizing the training images — a problem known as overfitting — a dropout layer with a rate of 0.5 is used before the final output layer. This layer randomly turns off some neurons during training, helping the model generalize better to new images it has not seen before.

The final layers consist of a dense layer with 256 neurons, which integrates features extracted from prior layers to make predictions, and an output layer that corresponds to the number of disease classes. The output layer uses a softmax activation function to predict the likelihood of each disease being present in the image.

During training, the model uses an Adam optimizer and monitors Top-1 accuracy metrics. This metric offer insights into the model's learning trajectory and its ability to generalize. Top-1 accuracy shows whether the model's first prediction is right. A low Top-1, suggests that the model has room for improvement regarding its confidence when selecting the correct classes. Monitoring this metric assists in identifying and rectifying issues like overfitting or underfitting, ensuring the model is learning effectively.



(a) Accuracy graph



(b) Loss graph

Figure 3.5: CNN accuracy and loss graphs on test set

The accuracy and loss plots across 50 epochs and batch size of 32, as illustrated in Figure 3.5, highlights the model's consistent training performance and its generalization on the validation data. These graphs showcase the model's robustness and reliability, with the accuracy graph showing synchronous improvements in both training and validation accuracies, and the loss graph indicating a decline in loss, which corresponds to an increase in model accuracy over time.

3.5.3.2 ResNet50

The deployment of the ResNet50 model takes advantage of transfer learning for the accurate identification and classification of diseases in vineyard images. As a model pre-trained on the ImageNet dataset, ResNet50 brings a deep and efficient architecture known for its effectiveness in image recognition tasks.

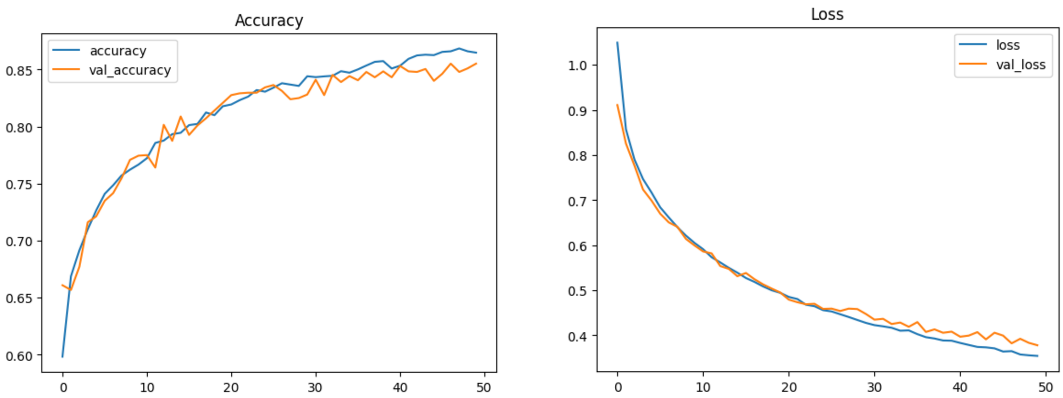
The ResNet50 model was loaded with its original, pre-trained weights, but without its top layers, making room for new classification layers designed for this specific task. The layers from the pre-trained model were frozen, so their weights stay the same during training. This approach transforms the model into a feature extractor, focusing training on the newly added layers.

On top of ResNet50, new layers were added to adapt the model to the vineyard disease classification. A Flatten layer was first added to convert the 2D features into a 1D vector, followed by a dense layer with 256 neurons and a ReLU activation function. This layer serves as a classifier on top of the comprehensive feature representations learned by ResNet50. Finally, an output dense layer with a softmax activation function was included, providing a probability distribution across the different disease classes.

The model was compiled with an Adam optimizer and trained for 50 epochs using a batch size of 32. The learning rate was set conservatively to ensure that the new layers learned from the vineyard images without disrupting the valuable features captured by the ResNet50 base.

Monitoring the accuracy and loss during training provided insights into the model's performance. The provided accuracy plot from the evaluation phase with the test set shows that the model could effectively distinguish between the different diseases, with the validation accuracy trailing closely behind the training accuracy, which suggests good generalization (see Figure 3.6a).

Similarly, the loss plot indicates that the error decreased significantly as the model learned, with both training and validation loss converging, a sign that the model was fitting well without overfitting (see Figure 3.6b).



(a) Accuracy graph (b) Loss graph

Figure 3.6: ResNet50 accuracy and loss graphs on test set

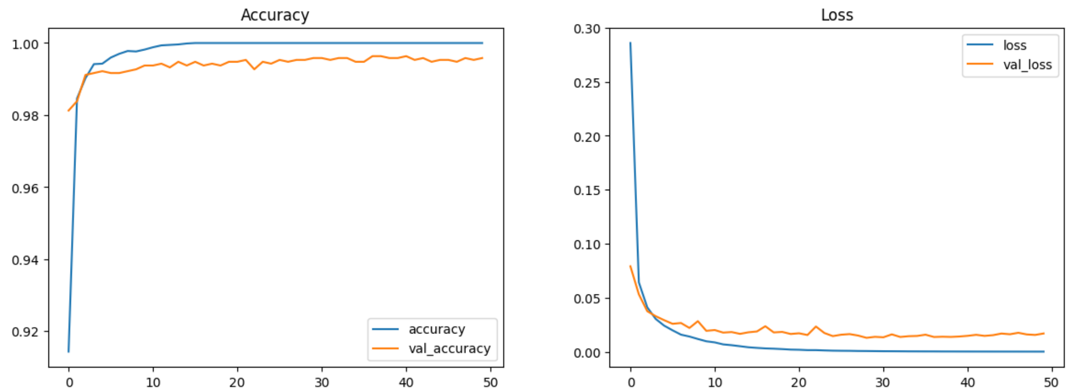
Additional layers on top of the ResNet50 model are essential to a transfer learning approach because they allow the model to move from recognizing general image features, which is what the pre-trained model is good at, to recognizing specific features important to the specific classification task at hand. These layers are where the learning of vineyard images occurs, adjusting the model output to the specific classes of interest in the dataset.

3.5.3.3 MobileNetV2

This model was adopted based on its efficiency and suitability for mobile devices. It is designed to make quick predictions and still be accurate. This model was loaded with its ImageNet weights for general feature detection, and the top classification layers were removed to make room for custom layers that are fine-tuned for the specific classification task.

Following the transfer learning approach, as with the ResNet50 model, the trainable parameters in MobileNetV2 were frozen to exploit the learned features while new layers were added for vineyard disease classification. These include a Flatten layer to convert 2D output to 1D vectors, followed by a dense layer with 256 neurons to learn complex features, and a softmax output layer corresponding to the number of disease classes.

The MobileNetV2-based model was compiled with an Adam optimizer and a reduced learning rate. This approach aimed to adjust the weights of the newly added layers carefully, preserving the valuable features inherited from the pre-trained model. It was trained over a period of 50 epochs, maintaining a consistent batch size of 32 images throughout the process.



(a) Accuracy graph (b) Loss graph

Figure 3.7: MobileNetV2 accuracy and loss graphs on test set

The model's performance is illustrated by the loss and accuracy plots in Figure 3.7. The loss graph displays a sharp initial decline, leveling off at a minimal value, which indicates the model's success in reducing error throughout its training. Meanwhile, the accuracy graph demonstrates that the model achieved and sustained high accuracy rates across both training and validation datasets. This consistent performance suggests that the model generalizes well and is not overfitting to the training data.

3.6 Satellite

This section delves into the integration of satellite technology with geospatial data via an API, highlighting its critical importance in improving agricultural practices. We explore the architecture and deployment of a system that utilizes the Flask API, emphasizing the interconnection among diverse environments and the employment of Convolutional Neural Networks (CNNs) for efficient data set retrieval and preprocessing. This groundwork enables advanced manipulation and representation of satellite and geospatial data, thus substantially improving agricultural methods by offering vital insights and visualization capabilities that prove particularly advantageous for agriculture. Additionally, this section discusses the combination of these datasets with the Flask API to improve the functionality of the application and user engagement, depicting the transition from data acquisition to its practical application in agriculture. The importance of Google Earth Engine's (GEE) vast satellite data repository is highlighted, showing its significant role in advancing agricultural practices through geospatial analytics.

Furthermore, the following section elaborates on the types and quality of satellite data accessible via the GEE API, essential for the system's support of agricultural endeavors. It examines the specifics of satellite imagery and geospatial data provided by GEE, such as resolution, frequency, and the assortment of datasets available, including vegetation indices, soil moisture levels, and land surface temperatures. Highlight how these data are crucial for precise agricultural monitoring, land use analysis, and environmental management. Methodologies for data retrieval, processing, and analysis are also discussed, with the aim of producing actionable insights for farmers, technicians, and researchers.

3.6.1 GEE API Connection

The API's minimalist design and the free-of-cost requests facilitate the seamless retrieval and processing of satellite imagery, allowing users to access, visualize, and analyze data efficiently. This connection is paramount for layer visualization, which is critical for agricultural insights, as it allows detailed observation and analysis of land use, crop health, and environmental changes over time. By leveraging GEE, the system can harness a vast repository of satellite imagery and geospatial datasets, offering a robust framework for agricultural decision making. The API's architecture is designed to accommodate the complexities of satellite data, ensuring a smooth interface for users to interact with and derive value from the geospatial analyses. This approach not only enhances the capabilities of the Flask API framework but also underscores the importance of satellite technology in advancing agricultural practices.

In addition to GEE, there are other possibilities of accessing Sentinel-2 data. There is the official Copernicus mission website [Ser23] and the corresponding guide [Ser14]. Using the browser provided by the website, it is possible to download a ZIP file with the data for a specific area. Normally you never have access to an image itself, it is a package that can be opened and worked with software such as SNAP or QGIS [Age; Tea02]. SNAP can be used to convert the data from that ZIP file to a GeoTIFF image based on several bands, and then that image can be viewed in QGIS and eventually for processing in other software. However, this process becomes laborious and costly in terms of resources. It forces you to somehow save the ZIP with the data or the GeoTIFF image so that it is always available for access, which increases the complexity of the code whenever you want to update the data to obtain the most recent information from the satellite.

There are other services that provide APIs based on data from Copernicus missions. Some are considered Data and Information Access Services (DIAS) by the Copernicus project itself [Com] and another well-known Sentinel Hub [Hub16]. These services provide pre-processed data and APIs in a more useful, simple, and direct way. The problem faced with many of the APIs was the limit of requests to services, where some of the services become paid after a few number of requests or after passing a trial test or sometimes having some services free but with limitations.

3.6.2 Data retrieved

Google Earth Engine tools provide access to a variety of satellite images and geospatial datasets that cover a wide range of applications. This includes high-resolution optical imagery, thermal imagery, radar imagery and a variety of derived data sets such as vegetation indices (e.g. NDVI to assess plant health), water index (e.g. NDWI for mapping water bodies) and soil moisture data. Mainly, the data is divided into three types: Climate and Weather data, Imagery data, and Geophysical data.

These datasets come from numerous satellite missions, each offering different resolutions, temporal frequencies, and spectral bands, allowing users to select the most appropriate data for their specific agricultural monitoring, environmental assessment, or land use analysis needs. In the decision-making process for selecting suitable satellite data to support agricultural applications, the Sentinel-2 mission, specifically the Multispectral Instrument (MSI), was chosen for its high resolution multispectral imagery. This data, available from 2015 to the present, is useful for a broad range of applications, including the monitoring of vegetation, soil and water cover as well as for humanitarian and disaster risk assessments. The utility and versatility of Sentinel-2 imagery make it an important resource for detailed observation and analysis of agricultural lands and environmental conditions.

The choice was further refined to utilize the Surface Reflectance data from the Sentinel-2 Level-2A product. This dataset provides orthorectified and atmospherically corrected surface reflectance data, available since March 28, 2017. The use of Level-2A surface reflectance data ensures that the imagery is ready for immediate analysis, removing the need for additional preprocessing steps to correct for atmospheric effects.

The code line provided below is an example of how to retrieve satellite data using the Google Earth Engine (GEE) API, specifically targeting the Sentinel-2 Surface Reflectance data:

```
collection = ee.ImageCollection("COPERNICUS/S2_SR_HARMONIZED").
                 map(maskClouds).filterBounds(polygon).
                 filterDate(str(one_weeg_ago),str(yesterday_date)).
                 filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE',20))
```

Listing 3.2: Satellite connection

The *filterBounds(polygon)* function limits the collection to images that intersect with a defined geographical area (land or parcel), ensuring that only relevant data are retrieved. By default, the system is configured to search for images within a time frame that starts 5 days prior to the current date and extends up to the day before the current date. This temporal window, referred to as the delta time, is crucial to ensure that the data collected is both relevant and timely for analysis purposes.

Each image in the collection contains several spectral bands, such as RGB (Red, Green, Blue) for basic color visualization, NIR (Near Infrared) for vegetation health assessment, and additional bands for water vapor and other environmental parameters. The choice of specific bands for analysis allows the creation of customized visualization layers customized to the particular needs of the application, such as assessing vegetation health, monitoring water bodies, or analyzing land cover changes. This selection process is essential to take advantage of satellite imagery for detailed and actionable insights into agriculture and environmental monitoring.

3.6.3 Visualization layers

Given the approach taken during the retrieval satellite data, this section and following subsections will detail the specific layers derived from satellite imagery that are instrumental for agricultural monitoring and chosen to be implemented. It serves as an introduction to the understanding and application of these layers, each tailored to highlight different aspects of the Earth's surface and its conditions.

Processing the retrieved data involves applying a mean to all images provided by the collection such as: `image = data.median().clip(polygon)`. This step results in a single image representing the average over a period (in this case, every 5 days), enhancing manageability and interpretability. Additionally, `.clip(polygon)` ensures that only data relevant to the defined land area (polygon) is retained for further analysis.

The layers processed and implemented through the Google Earth Engine (GEE) API and FlaskAPI provide invaluable insights into the health and condition of agricultural lands, water bodies, and other environmental features. By applying specific algorithms to the satellite imagery, we can extract detailed information on vegetation, moisture content, soil health, and water quality.

3.6.3.1 Satellite layer

This layer provides the base optical images captured by the satellite, offering a direct view of the Earth's surface. It is a critical component of remote sensing, is integral for a wide range of applications, from environmental monitoring and disaster management to urban planning and agricultural assessment. By capturing high-resolution images, the satellite layer enables detailed observations of land patterns, vegetation cover, water bodies, and human-made structures, providing invaluable insights for farmers and technicians.

```
    display = {"min":0, "max":5000, "dimensions":256,
                        "bands":['B4', 'B3','B2']}
    mapid = image.getMapId(display)
```

Listing 3.3: Satellite layer visualization

The following Figure 3.8 depicts the satellite layer view with a clear demarcation between natural landscapes and human habitation.

Figure 3.8: Satellite layer visualization

3.6.3.2 NDVI Layer

The Normalized Difference Vegetation Index (NDVI) is a widely used remote sensing index that offers a quantifiable measure of vegetation health, vigor, and biomass. It leverages the difference in the reflectance properties of vegetation in the red and near-infrared (NIR) parts of the electromagnetic spectrum. Healthy vegetation absorbs most of the visible light (including red) and reflects a large portion of the NIR, while stressed or sparse vegetation reflects more red light and less NIR [Ana24b]. The NDVI is calculated using the formula:

$$NDVI = \frac{(NIR + Red)}{(NIR - Red)} \tag{3.1}$$

Where NIR is the reflectance in the near-infrared spectrum, and Red is the reflectance in the visible red spectrum. NDVI values range from -1 to 1, where higher values indicate healthier and denser green vegetation. Very small values of -1 to 0, correspond to empty areas of rocks, sand or snow and possibly dead plants. Small values 0 to 0.3 indicate unhealthy plants. Moderate values from 0.3 to 0.6 represent healthy plants, shrubs and meadows. Large values range from 0.6 to 1 indicate very healthy plants or tropical forests. These values are a useful tool for determining whether fields are ready to be harvested; The lower the index, the closer a portion of the area is approaching the stage when it is ready to be harvested.

The NDVI layer is indispensable for monitoring crop conditions, changes in vegetation cover, and land use practices. It is especially useful for assessing photosynthetic activity and estimating biomass production. This capability makes it a critical tool for agricultural monitoring, environmental management, and the study of ecosystem dynamics [Ana24b].

The Python code snippet below demonstrates how to calculate the NDVI layer from an image, apply a color map for visualization, and prepare it for display:

```
#B8 is NIR, B4 is Red, B2 is Blue for visualization
image = image.select(['B8', 'B4','B2'])

#Calculate NDVI
ndvi = image.normalizedDifference(['B8', 'B4'])
colormap = create_colormap('RdYlGn',10)#Generate 10 colors

display = {"min":-1, "max":1, "palette":colormap,
           "dimensions":256}

mapid = ndvi.getMapId(display)
labels = create_colormap_labels()
```

Listing 3.4: NDVI layer visualization

This code first selects the necessary bands from the image (B8 for NIR and B4 for Red) and calculates the NDVI using the normalizedDifference function. It then generates a color map from red (representing low NDVI values, i.e., sparse or unhealthy vegetation) to green (representing high NDVI values, i.e., dense or healthy vegetation), which helps visualizing the vegetation health across the landscape. The display settings adjust the NDVI visualization parameters, including setting the minimum and maximum values of the NDVI scale and applying the generated color palette. Finally, the *mapid* is prepared for displaying the NDVI map, with labels providing a legend and additional context for the visualization.

This method of visualizing NDVI provides an intuitive way to assess vegetation health across large areas, making it an essential tool for environmental monitoring, agricultural management, and ecological research.

3.6.3.3 NDMI Layer

The Normalized Difference Moisture Index (NDMI) is designed to assess moisture content in vegetation and soil, leveraging the spectral properties of near-infrared (NIR) and

Figure 3.9: NDVI layer visualization

short-wave infrared (SWIR) light. Vegetation moisture content can significantly influence reflectance in these bands, making NDMI a valuable tool for monitoring water stress, planning irrigation, and efficiently managing agricultural resources. High NDMI values indicate high moisture content, while low values suggest dry conditions.

The NDMI is particularly useful for identifying areas at risk for drought or those experiencing abnormal moisture levels. By enabling the monitoring of moisture variations over time, it helps optimize water use and ensure sustainable agricultural practices. The index is calculated as follows:

$$NDMI = \frac{(NIR + SWIR)}{(NIR - SWIR)} \tag{3.2}$$

Where NIR represents the near-infrared band and SWIR represents the short-wave infrared band. These bands are selected based on their sensitivity to changes in water content, with NIR being absorbed and SWIR being reflected by water within vegetation [Ana24a].

Similarly to other indices, the NDMI operates within a range of -1 to 1. Negative values nearing -1 signal water stress, while values approaching +1 suggest waterlogging (see Figure 3.10). Consequently, each value within this spectrum corresponds to a distinct agronomic scenario.

For example, values from -1 to -0.8 signify barren soil; -0.8 to 0.0 represent low or very low canopy cover, indicating high water stress or low water stress, respectively. Values from 0 to 0.2 represent average canopy cover, reflecting high water stress or mid-low canopy cover under low water stress. Meanwhile, the range of 0.2 to 0.8 indicates mid-high to very high canopy cover with no water stress. Finally, values from 0.8 to 1 signify total canopy cover with no water stress or potential waterlogging.

It is crucial to recognize that NDMI values fluctuate throughout the growing season due to variations in plant reflectance across different phenological stages. Furthermore, there is a correlation between NDMI and NDVI. NDMI values indicating water stress can be confirmed by a significantly lower than average NDVI [Ana24a].

To implement the NDMI layer for satellite imagery analysis, the following Python code was implemented:

```
#B8 is NIR, B11 is SWIR, B3 is Green for visualization
image = image.select(['B8', 'B11','B3'])

#Calculate NDMI
ndmi = image.normalizedDifference(['B8', 'B11'])
colormap = create_colormap("Purples",10)#Generate 10 colors

display = {"min":-1, "max":1, "palette":colormap,
             "dimensions":256}

mapid = ndmi.getMapId(display)
labels = create_colormap_labels()
```

Listing 3.5: NDMI layer visualization

This script calculates the NDMI by selecting the appropriate bands, applies a color map for visualization, and prepares the imagery for display. The "Purples" color scheme is chosen to represent moisture levels, with darker shades indicating higher moisture content, thus facilitating the identification of water stress and optimizing irrigation strategies.

The ability of the NDMI layer to pinpoint variations in soil and vegetation moisture makes it a essential tool for agricultural management, drought assessment, and environmental monitoring. By providing information on water content across landscapes, it supports the effective allocation of water resources and the implementation of sustainable agricultural practices.

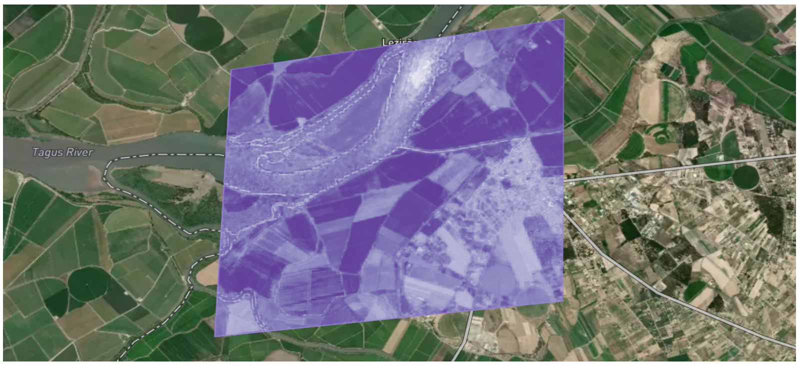


Figure 3.10: NDMI layer visualization

3.6.3.4 MSAVI Layer

The Modified Soil Adjusted Vegetation Index (MSAVI) is specifically designed to minimize the influence of soil background when detecting vegetation signals, which makes it particularly useful in areas where vegetation is sparse and soil has a significant impact on the observed satellite signal. The goal of MSAVI is to enhance the visibility of vegetation health and coverage by reducing the effect of soil characteristics that might obscure the satellite's view of the vegetation.

```
    #B8 is NIR, B4 is Red, B3 is Green for visualization
    image = image.select(['B8', 'B4','B3'])
    msavi = image.expression(
            '(2*NIR+1-sqrt(((2*NIR+1)**2)-8*(NIR-R)))/2',
            {
                'NIR': image.select('B8'),
                'R': image.select('B4')
            }
        )
    colormap = create_colormap('RdYlGn',10)#Generate 10 colors
    display={"min":-1, "max":1, "palette":colormap,
                "dimensions":256}

    mapid = msavi.getMapId(display)
    labels = create_colormap_labels()
```

Listing 3.6: MSAVI layer visualization

In the provided code snippet, the MSAVI is calculated using an expression that combines the Near-Infrared (NIR) and Red (R) bands from satellite imagery. The formula used is as follows:

$$MSAVI = \frac{(2 * NIR + 1 - sqrt((2 * NIR + 1)^2 - 8 * (NIR - R)))}{2} \tag{3.3}$$

This expression is designed to compute the MSAVI value for each pixel in the image, where the NIR band is represented by 'B8' and the Red band by 'B4' in the context of Sentinel-2 satellite imagery.

After the MSAVI calculation, a colormap is generated using the *create_colormap* function with the 'RdYlGn' color scheme, indicating a range from red (lower values, indicating less vegetation) to green (higher values, indicating more vegetation), with ten color gradations. This colormap is then applied to the MSAVI values to visually differentiate areas based on their vegetation health and coverage. The display parameters are set with minimum and maximum values of -1 and 1, respectively, and a fixed dimension size of 256 pixels, ensuring that the output is standardized for visualization purposes.

Lastly, the *getMapId()* function is used to generate a map visualization of the MSAVI data, applying the previously defined display settings, including the colormap. The *create_colormap_labels* function, generates labels for the colormap, providing a textual representation of the color gradations and their corresponding MSAVI values, further aiding in the interpretation of the map.

Focusing on mitigating soil background effects, MSAVI provides a more precise evaluation of vegetation health and coverage, especially in challenging environments where conventional vegetation indices may struggle due to significant soil background interference. This technique is most effective in the initial phases of plant growth.

Values ranging from -1 to 0.2 indicate bare soil, while 0.2 to 0.4 indicates seed germination and 0.4 to 0.6 denotes leaf development (see Figure 3.11). Once values exceed 0.6, NDVI becomes more appropriate, indicating dense vegetation cover obscuring the soil.

MSAVI is adept at detecting uneven sprouting, crucial for identifying vulnerable seeds recently sown, susceptible to predation, weather, or moisture damage. Implementing MSAVI enables the pinpointing of areas of poor germination, facilitating timely resowing to mitigate yield losses.

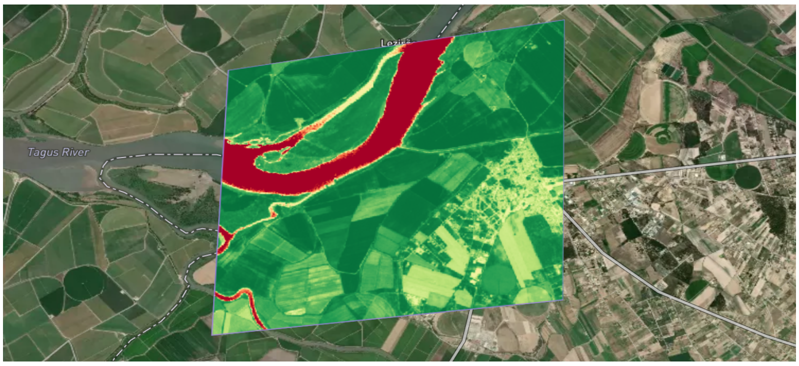


Figure 3.11: NSAVI layer visualization

3.6.3.5 NDWI Layer

The NDWI (Normalized Difference Water Index) layer is a critical tool for assessing water stress and quality within agricultural and natural ecosystems. It plays a vital role in managing water resources, pinpointing areas under water stress, and tracking the temporal changes in water bodies. Utilizing this layer enables a more nuanced understanding of water-related issues, facilitating strategic planning and interventions to address water scarcity and quality concerns.

The NDWI is calculated using the formula:

$$NDWI = \frac{(Green - NIR)}{(Green + NIR)} \tag{3.4}$$

Where *Green* represents the reflectance in the green band, and *NIR* is the near-infrared reflectance. This index is especially useful for emphasizing water bodies and distinguishing them from the surrounding land, due to the high absorption of NIR by water compared to the green light, which is reflected by water.

To implement the NDWI layer in Python, the following code can be used:

```
msavi = image.expression('(Green-NIR)/(Green+NIR)',
        {
```

```
            'NIR': image.select('B8'),
            'Green': image.select('B3')
          }
      )

colormap = create_colormap('PuOr',10)  # Generate 10 colors
display={"min":-1, "max":1, "palette":colormap,
            "dimensions":256}

mapid = msavi.getMapId(display)
labels = create_colormap_labels()
```

Listing 3.7: NDWI layer visualization

The code also includes steps for applying a color map to visually represent the NDWI values, enhancing the interpretability of the data for water resource management and conservation efforts. This approach to NDWI calculation and visualization is informed by methodologies and recommendations found in the literature and online resources, such as the EOS guide on their analysis platform [Ana24c].

Figure 3.12: NDWI layer visualization

3.6.4 Satellite limitations

Each of these visualization layers represent a powerful tool for the analysis and understanding of environmental and agricultural phenomena. By dissecting satellite data into these specific indexes and layers, farmers and technicians can gain deeper insights into the conditions of their areas of interest, facilitating informed decision-making and targeted interventions.

The approach taken in 3.6.2 is designed to balance the need for up-to-date information with the limitations imposed by the satellite's non-geostationary nature and the cloud cover that can obstruct the satellite's view of the Earth's surface. As a non-geostationary satellite, Sentinel-2 does not maintain a fixed position relative to the Earth's surface, and thus it cannot provide real-time imagery updates. Instead, images are captured as the satellite orbits the Earth, leading to a limitation by which data can only be fetched from the previous day or earlier. This temporal constraint necessitates planning and adjustments in data retrieval strategies for monitoring and analysis purposes, especially in applications that require up-to-date information. By focusing on images captured between 5 days ago and the day before the current date, the collection process prioritizes the retrieval of the most relevant and clear satellite imagery available within this defined window, maximizing the utility and applicability of the data for agricultural monitoring purposes.

The cloud masking function (*maskClouds*) and the subsequent filtering step that limits the dataset to images with less than 20% cloud coverage are essential to enhance the quality of the images retrieved. However, it is also important to recognize that on days with heavy cloud cover, this process may result in the exclusion of a significant portion of the images, potentially leading to instances where the returned image collection is sparse or even blank. This highlights the inherent challenges in relying on satellite imagery for real-time monitoring and analysis, necessitating alternative strategies or data sources in scenarios where cloud cover is a significant impediment. Addressing this challenge, a simple solution was implemented to enhance user interaction and data retrieval efficiency. When the cloud masking process and cloud coverage filters lead to instances where no suitable images are found within the default search window, the system is designed to issue an alert to the user. This alert signifies the absence of visible bands due to cloud obstruction, indicating the need for user intervention.

Users are given the option to adjust or increase the delta time - the period over which images are searched. By extending the search window beyond the default 5 days prior to the current date, users can potentially access a broader dataset, increasing the likelihood of finding images with minimal or acceptable levels of cloud cover. This user-driven approach allows for greater flexibility in data retrieval, catering to the specific needs and constraints of individual analysis scenarios. This feature underscores the system's adaptability and user-centric design, enabling more effective utilization of satellite imagery for agricultural and environmental monitoring. By allowing users to dynamically adjust search parameters in response to real-world challenges such as cloud cover, the system improves its utility as a robust tool for accessing and analyzing satellite data.

3.7 Weather

Weather plays a crucial role in agricultural productivity and environmental sustainability, influencing crop growth, water resource management, and ecosystem health. Given its critical importance, the integration of accurate and timely weather data into agricultural practices and environmental monitoring systems has become increasingly essential. This chapter delves into the mechanisms for acquiring and using weather data through an API connection, specifically focusing on Visual Crossing Weather Data & Weather API [Cro03].

The section begins by outlining the API connection (3.7.2), detailing the process of establishing a link to the Visual Crossing platform to access comprehensive weather datasets. Followed by the types of data retrieved (3.7.3), highlighting the diversity of available weather parameters and their relevance to agricultural and environmental analyzes.

The subsequent sections cover the technical aspects of making requests to the API (3.7.4) and the practical implementation of weather data in agricultural systems. These discussions encapsulate the methodologies for querying weather data, handling responses, and integrating this information into decision-making processes to improve agricultural productivity and environmental stewardship.

The chapter concludes by emphasizing the importance of weather data for agriculture (3.7.1). It outlines how accurate and timely weather information can empower farmers, technicians, and environmental scientists to make informed decisions, optimize resource use, and mitigate risks associated with weather variability.

3.7.1 Importance and usage

The importance of weather data in agriculture cannot be overstated. It plays an essential role in improving agricultural practices and promoting sustainable management of environmental systems [Bai+22]. This importance covers several aspects, including decision-making, operational planning, resource optimization, risk mitigation, improving sustainability and promoting technological integration in the agricultural sector:

- **Accurate and timely weather information** enables agricultural professionals to make strategic decisions related to planting, irrigation planning, pest management and harvesting operations. By predicting weather conditions, farmers can optimize the timing of these operations to improve crop yields, reduce resource waste, and minimize the risk of damage from unexpected weather events.

- **Advanced weather analysis** facilitates efficient water use, allowing irrigation planning in accordance with current weather forecasts and soil moisture. This optimization not only saves water but also ensures plants receive enough water at the most beneficial times.

- **Weather variability** poses significant risks to agricultural productivity. Extreme situations such as droughts, floods and heat waves can devastate crops, causing significant economic losses. Access to detailed weather forecasts and historical weather data allows for risk mitigation strategies, such as adjusting crop choices or using soil conservation techniques.

- **Sustainable agriculture** seeks to balance the needs of food production with the conservation of environmental resources. Weather data plays an important role in this effort by providing information on measures to reduce environmental impacts.

By adapting agricultural practices to weather conditions, agriculture can contribute to environmental conservation and resilience to climate change.

- **Integrating weather data with agricultural technology**, such as precision farming tools and decision support systems, marks a step forward in efficiency and effectiveness productivity of agricultural activities. Technologies that leverage weather analytics, such as automated irrigation systems and dynamic crop models, enable a data-driven approach to agriculture that supports higher yields, good crop health better and better economic results.

In short, the strategic use of weather data is an integral part of modern agriculture, offering solutions to some of the most pressing challenges facing the industry. As the climate continues to change, the reliance on accurate, detailed and timely weather information will only increase, highlighting the need for continued innovation and investment in weather analytics to support a sustainable future for agriculture and the planet.

3.7.2 API connection (Visual Crossing: Weather Data & Weather API)

Establishing a connection to Visual Crossing Weather Data & API is a straightforward process that opens access to crucial weather information for agricultural and environmental applications. This subsection details the steps involved in using Visual Crossing Weather API, a powerful tool for retrieving historical, current, and forecast weather data.

To begin, it is necessary to create a free account on Visual Crossing's platform, which grants the developer an API key essential for authenticating requests. This key acts as a unique identifier, ensuring secure access to the service. The API uses a simple, yet flexible request format that allows users to specify the location and time range for which they seek weather data. The basic form of an API request to the Timeline Weather API is as follows:

```
https://weather.visualcrossing.com/VisualCrossingWebServices/
rest/services/timeline/[location]/[date1]/[date2]?key=API_KEY
```

Listing 3.8: API connection request

In this URL, *[location]* can be a specific address, a city name, a latitude/longitude pair, or a postal code, offering versatile options for pinpointing the desired geographical area. The *[date1]* and *[date2]* parameters define the time range for the query, which can span from a single day to multiple years, depending on the user's needs.

By crafting a request URL with the appropriate parameters and the API key, it is possible to retrieve detailed weather data for their specified location and period. The Visual Crossing Weather API documentation provides comprehensive guidance on forming requests, understanding the response structure, and exploiting the full potential of the API. It offers examples, parameter descriptions, and troubleshooting tips, ensuring that users can efficiently integrate weather data into their applications and analyses [Cro03].

3.7.3 Data retrieved

The Visual Crossing Weather API returns data in a structured format, typically JSON, containing a wealth of information such as temperature, precipitation, wind speed, and much more, empowering users to analyze weather patterns, assess climatic conditions, and make informed decisions related to agricultural planning, environmental monitoring, and resource management. This comprehensive dataset encompasses various metrics crucial for agricultural decision-making, providing detailed weather information, including:

- Weather Data Arrays containing objects for each requested day, with day-specific weather data such as temperature, precipitation, and more. Each day object can contain an hour array, providing hour-by-hour weather details.

- Current conditions and alerts given at the time of the request, including temperature and other relevant metrics.

- Detailed weather data elements through properties such as cloud cover, conditions, temperature, humidity, and others, offering a full picture of weather at a given time. Other fields such as moon phase, sunrise, sunset, and moon rise provide additional context on celestial events.

- Statistical and forecast data give properties such as feels like, precipitation probability, and UV index that give insight into how the weather feels. The source attribute indicates the origin of the weather data, whether it is observational, forecast, historical forecast, or a combination thereof.

- Other metrics such as solar radiation and solar energy provide information on solar radiation and energy, crucial for applications in solar power generation and environmental monitoring.

This structured JSON response enables developers and analysts to integrate and process weather data within applications, providing end-users with detailed and actionable weather information tailored to specific locations and times. Each piece of data recovered is tailored to help farmers, technicians, and researchers understand current and forecast weather conditions. This information is crucial for planning agricultural activities, such as irrigation, planting, and harvesting, ensuring that operations are carried out at optimal times to maximize yields and minimize risks.

Visual Crossing Weather API offers a robust tool for accessing historical, current, and forecasted weather data. By leveraging this API, is possible to integrate precise weather information into agricultural management systems, enhancing its ability to make informed decisions based on the latest meteorological data.

3.7.4 Requests

Implementing API queries to retrieve weather data is an essential function for applications that require accurate and timely weather information. In the context of agricultural and environmental monitoring, two essential types of queries have been implemented: historical weather data retrieval and weather forecast data retrieval. These requests are supported through a Python-based backend that leverages the Flask framework to create API endpoints, which in turn communicates with the Visual Crossing Weather API.

3.7.4.1 Historical Weather Data

The *getHistoryWeatherData* function is designed to retrieve weather data from 14 days ago to the present day for a specified city. This historical weather data is important for analyzing trends, comparing and understanding past weather conditions that may have affected agricultural productivity or environmental conditions.

```
    response = requests.get(api_url)
    historyWeatherData[district] = response.json()
    historyWeatherData[district]['last_updated'] = date
    return historyWeatherData[district]
```

Listing 3.9: History Weather Data

This query involves constructing an API URL with the district name and the date range, then making a GET request to the Visual Crossing Weather API. The response, formatted as JSON, is processed and stored, with the *last_updated* timestamp ensuring the data's freshness.

3.7.4.2 Weather Forecast Data

The getForecastWeatherData function is intended to provide weather forecast information for a specific city. These forecast data are essential for planning future agricultural activities, predicting weather-related risks, and taking proactive measures to mitigate adverse impacts on crops and livestock.

```
    response = requests.get(api_url)
    weatherData[district] = response.json()
    weatherData[district]['last_updated'] = date
    return weatherData[district]
```

Listing 3.10: Future Weather Data

Similarly to historical data retrieval, this function constructs a request to the Visual Crossing Weather API and processes the JSON response. The data includes a wide range of forecast weather metrics, which are then made available for further analysis and display.

To make these weather data retrieval functions more accessible from the front-end application, Flask API endpoints were implemented. These endpoints allow for dynamic retrieval of historical data and weather forecasts based on user requests from the frontend. The data can then be displayed as tables or graphs, allowing users to interact with and analyze weather information relevant to their agricultural or environmental needs. This integration of weather data through Flask API endpoints illustrates the power of combining weather analytics with web technology to support data-driven decision making in agriculture and beyond.

4 Prototype

This section focuses on the design and development of the website SmartData, as well as its integration with an already developed and tested application named AgriDash. The prototype developed for SmartData allows the creation and monitorization of agriculture fields using the architecture defined in section 3.1. The SmartData frontend was developed to meet the specifications detailed in section 1.3 and to showcase the application use cases discussed in section 3.4.

This chapter details the development and functionalities of the prototype application its design decisions and available features, giving extra focus on the integration architecture of the resultant application, available features, such as screens, navigation properties and, data management.

4.1 SmartData

In this section, we delve into the SmartData platform, a key component of our integrated digital agriculture solution. SmartData is designed to empower farmers and agricultural professionals with the tools needed to efficiently create, manage, and monitor agricultural fields. It serves as the interface for our comprehensive agriculture management ecosystem, integrating seamlessly with AgriDash, a previously developed application, to provide a holistic view and management capabilities for agricultural operations.

4.1.1 Design decisions

The development of SmartData was guided by several key design decisions aimed at fulfilling the project's vision, as described in section 1.3, and effectively addressing the use cases discussed in section 3.4. These decisions were instrumental in shaping the platform's architecture, user interface, and functionality.

User-Centric Design

SmartData's interface was created with the end user in mind. Aware of the varying technological skill levels of the target user base, the platform has an intuitive and easy-to-navigate user interface. This design choice ensures that users can easily access and use the platform's features without requiring in-depth technical knowledge, thereby expanding accessibility and usability of platform.

Integration with AgriDash

A key design decision was the seamless integration with AgriDash. By leveraging and extending AgriDash capabilities, SmartData provides a unified platform that not only facilitates the creation and monitoring of agricultural fields but also improves decision making through analytics comprehensive data. The integration architecture has been carefully designed to ensure data consistency, real-time synchronization, and a consistent user experience across both applications.

Scalable Architecture

Understanding the dynamic nature of agricultural management and the potential for future expansion, the platform has been developed with a scalable architecture. This approach allows adding new features, integrating new data sources, and meeting the needs of a growing number of users without compromising performance. The architecture is detailed in the 3.1 section, using modern development methods and technologies to ensure flexibility and scalability.

Focus on Data Management

SmartData emphasizes effective data management. The platform supports the creation, modification and visualization of agricultural field data, providing users with the necessary tools to manage their operations effectively. Features such as real-time data updates, comprehensive analytics, and good visualisation techniques are an integral part of the platform, allowing users to make informed decisions based on accurate information and update.

4.1.2 Available Features

SmartData is equipped with a range of features designed to meet the specific needs of the agriculture sector.

4.1.2.1 Satellite View

SmartData's Satellite View function provides a powerful and intuitive tool to view agricultural fields from a panoramic perspective, using advanced geospatial technologies to improve agricultural management, as seen in Figure 4.1. Leveraging high-resolution satellite imagery, this capability is critical for monitoring crop health, assessing field conditions, and accurately planning agricultural operations. Users can obtain up-to-date visual data about their field, allowing for detailed inspection and informed decision-making.

Figure 4.1: Satellite view

This feature integrates Mapbox [Map10] for visualization and map creation, utilizing the "satellite-v9" style provided by the API. This style is created from images from various satellites, which does not guarantee its currency. To address this, SmartData provides the option to view the terrain directly from the Sentinel-2 satellite, ensuring more up-to-date images for users.

Map controls are visibly positioned in the upper right corner, allowing users to engage in a rich map experience through a full-screen view, and to create or delete polygons directly on the map. This improves user interaction and spatial analysis within the app.

On the opposite corner, options for viewing satellite layers are displayed, along with toggle buttons for markers. This layout helps accessing different data layers and visibility markers, allowing users to customize their viewing experience based on specific needs or preferences.

This integration of Mapbox, combined with user-friendly map controls and layer display options, demonstrates SmartData's commitment to providing advanced tools for agricultural management. With these features, SmartData ensures that users have access to the most relevant and up-to-date geospatial information to make informed decisions in their agricultural operations.

4.1.2.2 Field Creation and Management

The Field Creation and Management feature is designed to allow users to easily define and manage their agricultural fields within the platform, provides a comprehensive set of tools for drawing field boundaries and manage field data over time, figure 4.2. Through its intuitive interface, users can locate fields directly on the satellite view, ensuring an accurate and up-to-date representation of their agricultural lands. This functionality is essential to support efficient agricultural operations and data-driven decision making.

In addition to the basic capabilities, the feature includes several advanced functions for field management:

- **Polygon Editing:** Once polygons are drawn to represent field boundaries, they can be modified. This allows users to adjust their field shape as needed to accurately

Figure 4.2: Agriculture field creation

reflect boundaries as seen in Figure 4.3b.

- **Multiple Polygons with Distinct Borders:** Users can draw multiple polygons, each delimited by a border of a different color. This visual distinction makes it easy to identify separate fields or areas within a view as seen in Figure 4.3a.
- **Polygon Selection:** It is possible to select from the different polygons drawn. This functionality is essential for managing multiple fields or parcels simultaneously, providing a streamlined way to focus on specific segments as seen in Figure 4.3c.
- **Polygon Dragging:** Users have the ability to drag polygons across the satellite views. This feature allows field boundaries to be repositioned without having to be redrawn, providing flexibility in managing field layouts as conditions or requirements change.

(a) Multiple drawn polygon. (b) Select polygon. (c) Edit polygon.

Figure 4.3: Field management features

With accurate field delineation, easy editing, and clear visual field differentiation, the platform ensures that users can efficiently manage their agricultural resources with up-to-date and accurate geospatial data.

4.1.2.3 Satellite Layers Visualization

The Satellite Layers Visualization feature is a vital component of SmartData, allowing users to overlay various satellite-derived indices on their agricultural fields. This functionality is exemplified by the MSAVI (Modified Soil Adjusted Vegetation Index) layer, a key index for assessing soil and plant health, section 3.6.3.4. The MSAVI layer, shown in Figure 4.4, helps in minimizing soil background effects on vegetation signals, providing farmers and viticulturists with a clearer understanding of crop vigor and condition. With this feature, users can base their decisions on a comprehensive analysis of satellite data, leading to more informed and effective agricultural practices.

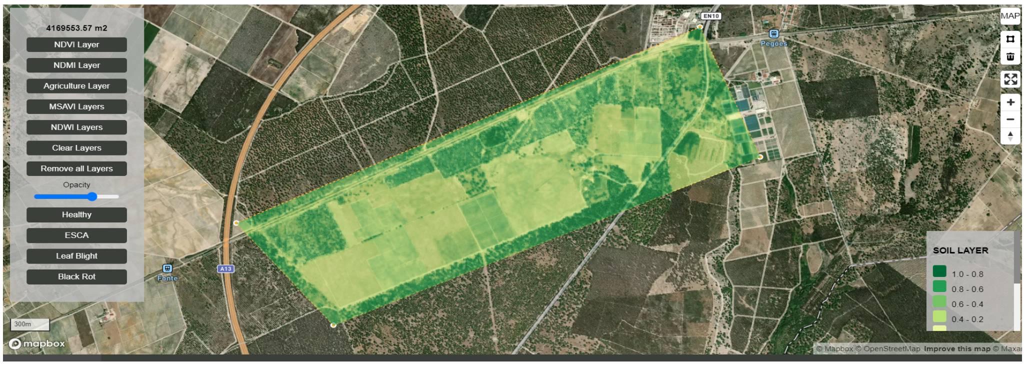


Figure 4.4: Visualization of MSAVI
(Modified Soil Adjusted Vegetation Index) layer

The image provided showcases an agricultural field with the MSAVI layer applied. The visualization reveals various shades of green, indicating the vigor and health of the vegetation. A legend on the right side of the image correlates the color intensity with the index value, aiding in the interpretation of the data.

Key functions of the Satellite Layers Visualization feature include:

- **Opacity Control:** Users can adjust the opacity of the applied layer, allowing for a customized view that can reveal both the satellite index information and the underlying field details.
- **Dynamic Legend:** A legend is displayed and dynamically updated whenever a polygon with an applied layer is selected. This helps users interpret the index values represented on the map.
- **Multiple Layer Visualization:** There are five different visualization layers available, each providing distinct insights into agricultural conditions. Section 3.6.3 for the implemented layers.
- **Layer Management Buttons:** Two buttons are provided for managing the layers on the selected polygon. One button clears the cache to remove layers, ensuring that outdated data does not interfere with current analysis. The other toggles the visibility of the layers, allowing users to view the underlying satellite imagery without indices.
- **Resource-Efficient Caching:** Layers are stored in a cache system, conserving resources from the Google Earth Engine (GEE) API and providing faster, easier access to information.

The complex layer management and visualization system as shown in Figure 4.4 plays a significant role in modern precision agriculture. It allows for fast, efficient and resource-efficient data analysis, providing agricultural professionals with timely access to valuable information for optimal field management.

4.1.2.4 Weather Data Visualization

The ability to visualize weather data plays a significant role in agricultural planning and management. The Weather Data Visualization feature in SmartData allows users to graphically represent various weather parameters over time. As shown in the provided image 4.5, this feature includes a chart that presents temperature data for a specific location over a period of days in October.

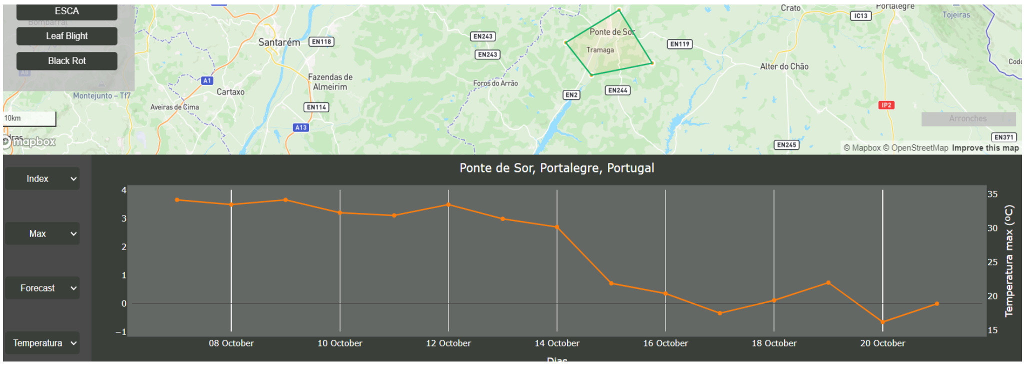


Figure 4.5: Graph for weather data visualization

The graph in the image illustrates temperature changes over a span of two weeks, with the line graph clearly showing the trend. Each data point represents one day, with arbitrary vertical lines representing the range between maximum and minimum temperatures for that day. Users can use this detailed weather information to make informed decisions about planting, irrigating, and preparing for potential extreme weather events, ensuring the resilience and productivity of their agricultural operations.

This feature employs reverse geocaching to determine the closest city to the center coordinates of the selected polygon. It then utilizes the Mapbox API to retrieve the city name, which is then used in an HTTP request to fetch weather data for that city (Section 3.7.2). Users can choose between forecasted and historical weather data, enabling them to plan for future conditions as well as analyze past weather patterns.

Both forecast and historical data are available for a span of 15 days, and the interface allows users to zoom in for a more detailed view of each day's data. Within these options, it is possible to select different types of data, such as temperature, humidity, wind speed, and precipitation. Users can also choose to view the maximum, average, and minimum values of these weather parameters.

4.1.2.5 Disease Classification

The introduction of the Disease Classification feature in SmartData represents a notable development in the field of agricultural technology. It employs image recognition and machine learning algorithms to identify and classify common crop diseases from visual data, figure 4.6. This tool empowers farmers with the capability to swiftly detect diseases facilitating early intervention which is key to preserving crop quality and yield. The application of this feature illustrates the convergence of AI and precision agriculture, providing cutting-edge solutions for disease management.

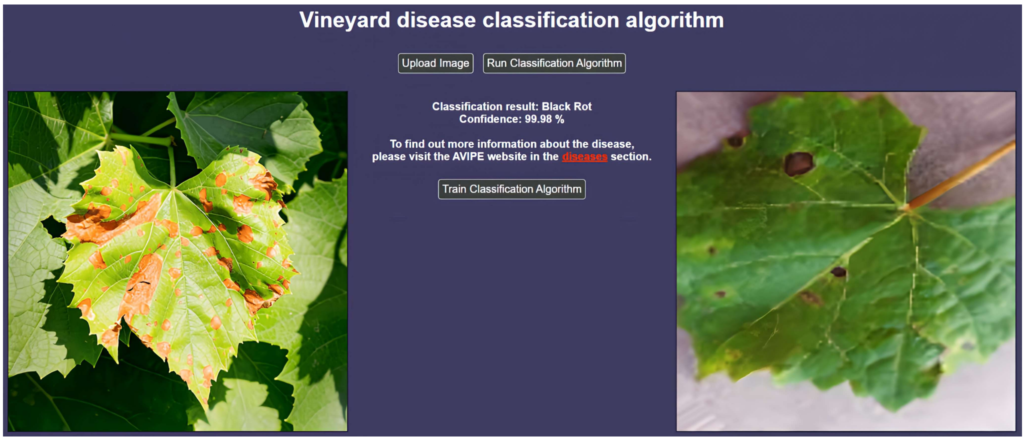


Figure 4.6: Disease Classification

When uploading images of their crops, farmers receive immediate visual feedback as the images appear on the left side of the interface. At the heart of this feature is a complex classification algorithm, chosen for its effectiveness, as detailed in section 3.5. It examines uploaded images, distinguishing patterns and abnormalities consistent with known signs of disease.

The results of this analysis are presented to the user in a clear and concise manner. In the central part of the screen, a text detailing the classification result—namely, the disease label—emerges, accompanied by the confidence level expressed as a percentage. This pivotal information is supplemented with additional text that, when interacted with, directs users to the AVIPE website for a deeper understanding of Portuguese grapevine diseases [AVI21].

To the right, the user is aided further by the inclusion of a reference image that corresponds with the identified disease label. This image, sourced from the database that was used to train the model, serves as a visual guide, allowing users to make a side-by-side comparison with their uploaded image, thereby reinforcing the accuracy of the diagnosis.

Moreover, the tool offers an advanced feature for technicians: the option to refine and train the classification model with the new image data provided during the upload. This aspect of the tool ensures that the underlying algorithm continues to evolve and adapt, enhancing its diagnostic precision with each new image analyzed. However, in recognition of the varying levels of technical expertise, this advanced functionality is reserved for technicians and is accessible exclusively via the web version of the application, underscoring a tailored approach to user access and experience. It is worth noting that suggestions for refining the model are registered to be trained later along with several requests, rather than undergoing immediate re-training.

SmartData's Disease Classification tool embodies the integration of AI into agriculture, providing innovative solutions for disease management that are not only accurate but also user-centric, ensuring that farmers and agricultural professionals are equipped to maintain crop health and optimize yield effectively.

4.1.2.6 Markers and other functionalities

The Markers and other functionalities section of SmartData reveals the application's advanced capabilities in enhancing agricultural monitoring with precision. The images provided have two main aspects: the disease markers based on classification data figure 4.7, and the dynamic representation of satellite views figure 4.8.

Figure 4.7: Geospatial visualization of vineyard health.

In SmartData's disease classification feature, once a machine learning algorithm identifies a disease, metadata from the classified image is used to create a geospatial marker on the map. This marker is more than just a static icon; it comes to life when interacted with, displaying a pop-up window. The pop-up window contains the image, date the image was taken, and title with the disease label assigned during triage. There is also a "Treatment" button, which when clicked will remove the marker and confirm that the disease or problem has been resolved. This action-oriented feature supports agricultural workflows, moving from detection to action seamlessly.

The user interface is intuitive, with buttons in the top left corner of the map representing the various diseases that can be marked. These serve as toggles to show or hide markers related to each disease, with different colors providing an immediate visual distinction between disease clusters. Such functionality is essential for farmers in monitoring many large-scale outbreaks.

The second feature highlighted is the tilted map view, which offers a 3D perspective of the terrain. By providing this angled satellite view, users will gain a better understanding of the agricultural landscape, which can be important for planning and monitoring purposes.

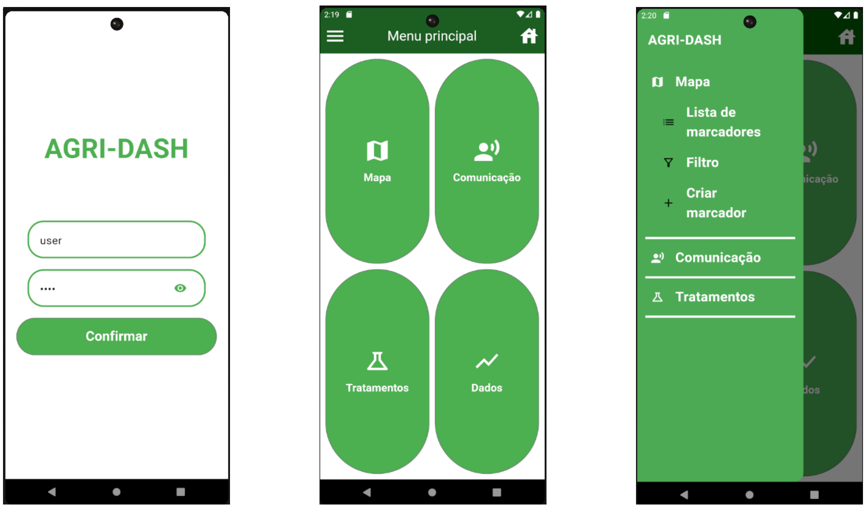
Figure 4.8: Tilted Map View

Elevations and contours come into play in this angular view providing contextual depth that a flat map cannot, making it an essential tool for those who need to understand the nuances of the terrain image.

This innovative integration of markers and tilted map views highlights SmartData's commitment to provide comprehensive, user-friendly and useful agricultural information that combines technology and practicality practices for effective agricultural management and disease control.

4.2 AgriDash Integration

The integration of SmartData with AgriDash signifies a perfect union of data collection, visualization and analysis relevant to precision agriculture. AgriDash, a tool previously developed to leverage various data methods for agricultural management, aligns with the goals of this project by providing a user-centric platform designed with Human-Computer Interaction (HCI) techniques.



(a) Login. (b) Menu. (c) Side bar menu.

Figure 4.9: AgriDash interface

This section explores the collaboration features between the two platforms, illustrating how data synchronization and analytics can improve agricultural operations. The co-creation approach, involving farmers and technical advisors, has been fundamental to AgriDash's development, allowing it to effectively respond to specific user needs. Its interface, as shown in Figure 4.9, emphasizes user-friendly interaction.

Within the AgriDash application, this integration effort will focus on the "Data" section as shown in the menu (Figure 4.9b). This area of the application will be especially important as it will be expanded to display a suite of data analytics and visualizations needed to enable farmers and technicians to make data-driven decisions. The integration will ensure that the "Dados" section has access to the rich dataset provided by SmartData, allowing for comprehensive monitoring and analysis of agricultural data.

This collaborative ecosystem will effectively report issues and promote conditions through a backend API, converting them into useful notifications for users, thereby simplifying the decision-making process. AgriDash will interact with this API to retrieve and display analysis results, ensuring that farmers and advisors have access to relevant data visualizations.

4.2.1 Architecture Integration

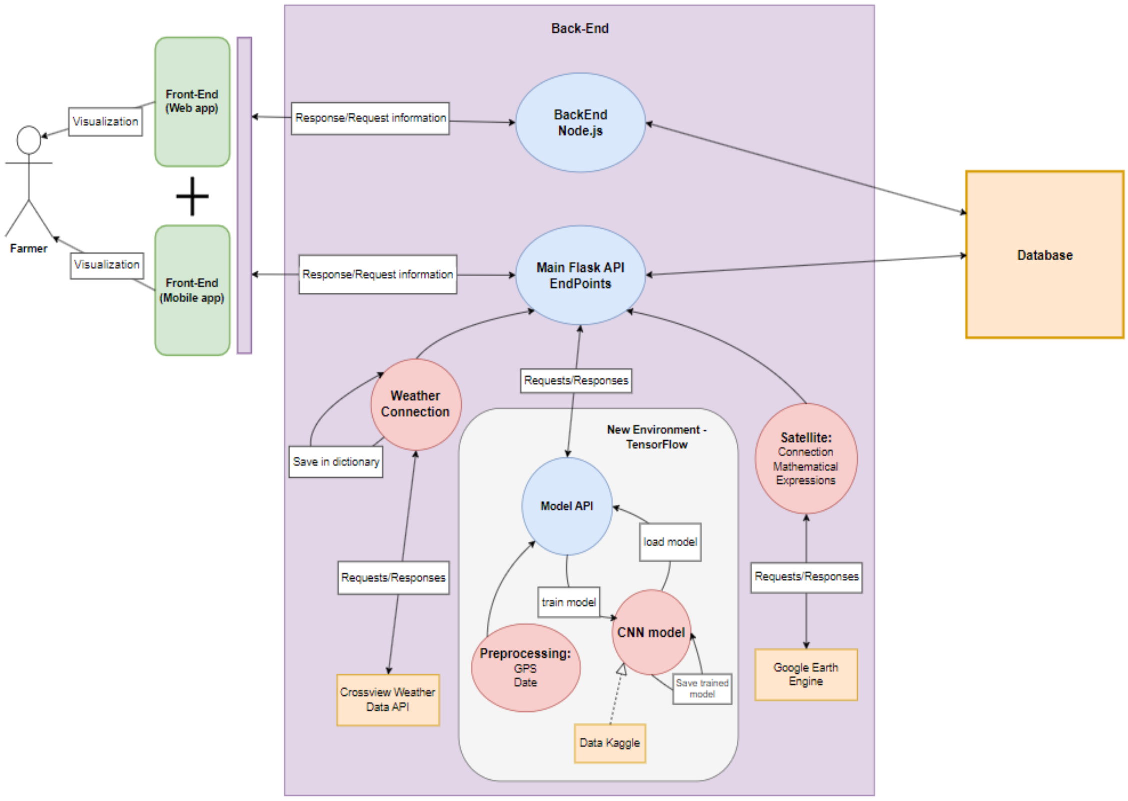


Figure 4.10: Architecture of the integrated solution.

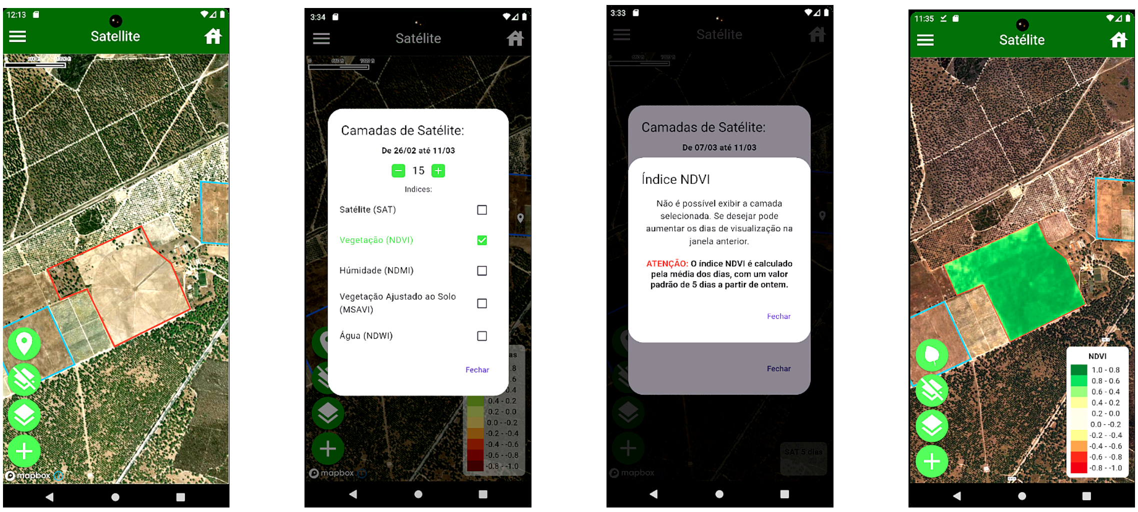
Combining the robust features of SmartData with AgriDash has created a complex yet efficient system architecture. The core of this integration is depicted in the Figure 4.10 above, where both platforms' backend operations—SmartData's Flask API and AgriDash's Node.js—are connected to a shared database. This ensures seamless data flow and real-time updates, facilitating a dynamic and interconnected experience for the user.

AgriDash, initially developed using Flutter for frontend development and Node.js for the backend, posed a unique challenge in integrating features with SmartData. This required the continuation of the existing frontend version for AgriDash in Flutter, ensuring the replication of SmartData's functionalities while preserving its own identity and user experience.

The result is a comprehensive architecture that combines SmartData's real-time field monitoring and management capabilities with AgriDash's analytics capabilities, creating a unified system that meets the diverse needs of modern agriculture. This integrated approach not only streamlines operations but also paves the way for future improvements and scalability.

4.2.2 Satellite Screen

Following the integration of SmartData's robust features into the AgriDash application, the Satellite Screen section highlights an important development, providing users with an advanced interface to analyze satellite data within the system. Integrating these features as part of AgriDash represents a smooth combination of technologies, enhancing application functionality and user experience.



(a) Satellite View. (b) Satellite Layers. (c) Layer Exception. (d) NDVI layer

Figure 4.11: AgriDash Satellite

The satellite view section within AgriDash is a significant improvement over its predecessor SmartData, providing users with a more personalized and efficient experience. Upon logging into the AgriDash application and selecting the satellite option, the interface immediately shows the user their associated agricultural parcels. This custom view, as depicted in figure 4.11a, is the result of the Flask API fetching information from the database regarding the lands owned by the user. These plots are initially highlighted in blue for easy identification.

Improving the human-machine interaction, when a user selects one of their plots, the interface visually changes the border color to red. This visual cue is not just for aesthetics; it is a thoughtful design choice that simplifies the identification of active selections and assists users in navigating through their lands within the application.

Moreover, in keeping with the need for precise and directed actions, the Satellite View in AgriDash restricts the interaction with layers to only when a plot is selected. Should a user attempt to engage with the layers without a selected polygon, the application will proactively respond, issuing a warning. This ensures that users are reminded to select a polygon first, streamlining the process and maintaining a focus on specific areas for analysis or intervention. This approach avoids the potential confusion of interacting with multiple layers without a clear context, thereby maintaining a structured and user-friendly environment.

The satellite layers feature represented in figure 4.11b is further enhanced by a user-controlled temporal aspect. While users have the ability to overlay different indices (section 3.6.3) to gain valuable insights into crop health and environmental conditions, they are also afforded the flexibility to manipulate the time frame for which these layers are displayed.

At the top of the layer selection, two green buttons allow users to adjust the display of satellite data to show only the data from the last x days, rather than selecting specific date ranges. This temporal control is crucial for users who need to analyze trends over recent periods or focus on specific time frames that may correspond to important events or anomalies in agriculture.

By offering the ability to adjust the duration for which indices are presented, AgriDash provides users with a more dynamic and customizable tool for precision agriculture and sustainable resource management. This feature not only aids in the immediate assessment of current field conditions but also in historical analysis, providing a more comprehensive understanding of a crop's growth and response to environmental factors over time.

The layer exception handling feature within AgriDash, as depicted in figure 4.11c, is an innovative response to the limitations introduced by satellite imagery, specifically those discussed in section 3.6.4. This mechanism is important to ensure the reliability and usability of satellite data within the application, especially when satellite images are obscured by clouds.

When users encounter a satellite image in which land is not visible, the layer exception handling is triggered. This event occurs if it is not possible to select bands in the image that accurately represent the parcel, a situation in which even removing clouds does not yield a usable image.

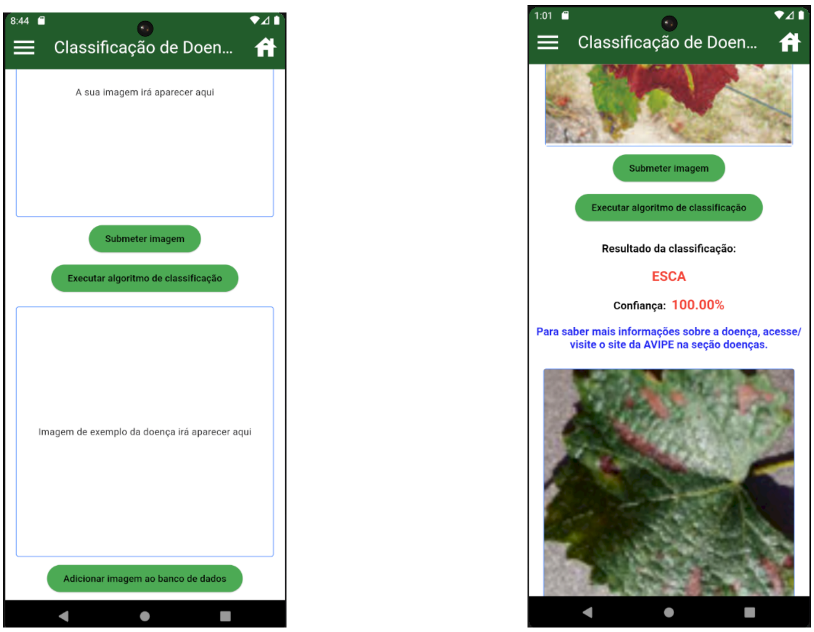
This proactive approach to handling exceptions ensures that users are not left without options or guidance. By recognizing the inherent challenges in using satellite data and providing clear, actionable responses, AgriDash reinforces its role as a valuable tool for modern agricultural management, enabling users to navigate around natural limitations and maintain continuity in their data analysis and decision-making processes.

Lastly, the NDVI Layer feature, shown in figure 4.11d, allows for a detailed view of vegetation health within the agricultural fields. By assessing the density and health of crops, users can optimize their cultivation practices, enhancing yield and reducing the risk of crop failure.

Each of these figures represents a snapshot of the comprehensive features available in the AgriDash application, following its integration with SmartData. The architecture of this integrated solution, detailed in section 4.2.1, facilitates the interoperability between SmartData's detailed satellite imagery functionalities and AgriDash's user-focused analytical tools. This powerful combination in a single application environment provides a valuable tool-set to the agricultural industry, ensuring accuracy and ease of use for all stakeholders involved.

4.2.3 Disease Classification Screen

The disease classification screen in AgriDash, shown in figure 4.12, mirrors the disease classification feature of SmartData (section 4.1.2.5), providing equivalent functionalities and operating in a similar manner. This continuity ensures that users familiar with SmartData will find the AgriDash environment intuitively accessible, maintaining consistency across platforms.



(a) Classify view. (b) Classification.

Figure 4.12: AgriDash Disease Classification

Upon navigating to this screen, users can upload images of their crops for analysis. Images are placed prominently at the top of the screen, making them easy to reference.

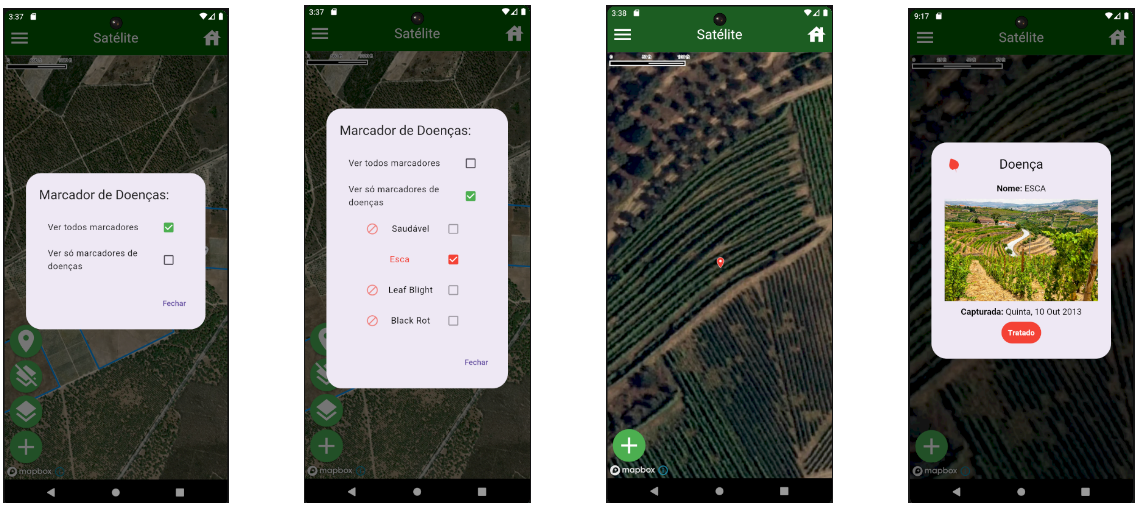
Once the classification algorithm processes the image, the diagnosis result is displayed as text below the image. Accompanying the textual result is a representative image corresponding to the disease label attributed to the uploaded photo.

A distinct feature that improves the user experience is the interactive blue text, which, when selected, redirects users to the AVIPE disease database [AVI21]. This link provides users with a wealth of information, allowing them to better understand the specific disease affecting their crops and explore potential treatments or management strategies.

This integration between AgriDash and the AVIPE database exemplifies the application's commitment to providing comprehensive solutions for disease management in agriculture. By empowering users with immediate classification results and resources for further education, AgriDash positions itself as a critical tool for farmers aiming to safeguard and improve the health of their crops.

4.2.4 Markers Screen and other functionalities

The Markers Screen and other functionalities in AgriDash is closely parallel with the section 4.1.2.6 previously discussed for SmartData. In AgriDash, users can visualize all markers that pertain to their landholdings, which could represent various conditions, not limited to diseases (for example tractors, water points etc.).



(a) Markers menu. (b) Markers menu. (c) Disease Marker. (d) Marker popup.

Figure 4.13: AgriDash Markers

The markers menu, as shown in Figures 4.13a and 4.13b, illustrates the application's interactive list of potential issues that can be tagged on the map, such as different diseases. The options presented are directly relevant to the user, displaying only the markers that are available in their plots. If a particular disease marker is not present on the user's database, the option to view or toggle that marker is disabled, thus providing a clutter-free and focused user experience.

The simplistic red marker depicted in Figure 4.13c exemplifies a disease marker, in this case for ESCA, placed on the map. It serves as a straightforward and distinct visual cue for users to identify and locate specific concerns within their fields.

Figure 4.14: Tilted Map View.

Upon selecting a marker, a detailed popup appears as illustrated in figure 4.13d. This popup contains information such as the image of the affected area, the date of the recorded issue, and of course the name of the present disease. The popup is an essential feature that provides context and facilitates the decision-making process regarding disease treatment or further investigation.

Lastly, AgriDash includes an "extra function" as shown in Figure 4.14, which allows for a tilted map view similar to SmartData's 3D representation. This immersive view enhances the understanding of the terrain and the spatial distribution of markers. Markers in this mode are rendered in 3D, providing a more clearer sense of the location and extent of the problems represented by the markers.

AgriDash's approach to markers and functionality integration extends the practical and analytical capabilities of SmartData, focusing on an intuitive and streamlined user experience that is both responsive and contextually aware. This alignment of features across platforms ensures users can transition smoothly between SmartData and AgriDash, enjoying a consistent and powerful tool-set for agricultural management.

5 Evaluation and Results

This chapter presents the results and the assessments from evaluating the performance of the developed models and the final application through two distinct phases. The first phase, focuses on the technical evaluation of the models, where various performance metrics are evaluated. This phase employs a thorough approach to test the models' abilities to classify vineyard diseases accurately. The second phase redirects the focus to the practical implementation of the models in an application aimed at the end user, bringing the technology into the hands of those who will use it directly in the field. It is important to highlight that, at this stage, feedback was rigorously collected and valued, especially from a key user, a technical agronomist from AVIPE, who played a critical role in testing the application and providing professional insights. Application functionality, usability and overall end-user satisfaction were assessed with his essential contribution, ensuring that specific stakeholder needs and expectations were met and refined as necessary.

The process of testing will be detailed in the section 5.1. The models, initially developed and trained, will be subjected to a set of tests using both the validation and test datasets. It involves evaluation of the CNN models, encompassing the custom CNN, the ResNet50, and MobileNetV2 implementations based on various metrics such as accuracy, precision, and f1-Score. Additionally, this section will set the stage for the end users' tests, providing an overview of how these tests will be conducted, the demographic of the test group, the metrics for evaluation, and the methodology for collecting and analyzing feedback.

Following the evaluation, the section 5.2 will present a thorough analysis of the results. Dedicated to present the findings from both phases of evaluations. It will compare the performance of the various models, interpret the importance of the results, and discuss their implications for the field of precision agriculture.

5.1 Evaluation

The evaluation of CNN models and the end-user application constitutes the main focus of this dissertation and is essential to understand their effectiveness and impact in the real world. This section describes the evaluation of convolutional neural network (CNN) models designed for the classification of vineyard diseases and the subsequent usability tests of the application that incorporates these models.

5.1.1 Model evaluation

The evaluation of the CNN models is organized by providing a detailed analyzes of their performance. This evaluation will cover the three models described in the 3.5.3 section: a custom-designed CNN, the modified ResNet50, and the modified MobileNetV2. Each model will be tested using the same criteria to ensure comparability of the results.

Each model will be evaluated on a test set made up of unseen vineyards images. This serves as an indicator of the accuracy of their classification in real-world scenarios.

Validation techniques were implemented to evaluate the stability and reliability of the models. This step is designed to protect against overfitting and ensure that the models are robust across multiple subsets of data.

The performance of the models was quantified using the main evaluation metrics: **Accuracy**, **Precision**, **Recall** and **F1-Score**. These metrics were used to evaluate the models both after the direct tests on unseen images and after the cross-validation steps to ensure that the models performance is reliable and consistent across multiple scenarios.

5.1.1.1 Test set

Each model will be evaluated using a distinct test set that the models have not encountered during their training. This ensures an unbiased assessment of their performance. The test set is a collection of vineyard images that represents 20% of the original data set (see section 3.5.2). This dataset will serve as the benchmark for the models' ability to generalize and perform in real-world scenarios similar to those that is encounter post-deployment. The following code represents the steps taken to evaluate the models in the test set.

```
def evaluate_model(mod, test_images):
    results = mod.evaluate(test_images, verbose=0)

    print("Test_Loss:␣{:.5f}".format(results[0]))
    print("Accuracy_on_the_test_set:␣{:.2f}%"
                    .format(results[1] * 100))
    print(f"Top-1_Accuracy:␣{results[2]}")
```

Listing 5.1: Model evaluation function on test set

5.1.1.2 Cross-Validation

Cross-validation (CV) is an important technique in machine learning to evaluate a model's generalization capability, or its ability to perform consistently across different sets of data. This concept reflects the human ability to recognize patterns and objects, such as identifying a dog even if we have never seen that breed before. However, achieving similar generalization in machine learning models can be challenging, it needs rigorous evaluation. Cross-validation achieves this goal by allowing the model's performance to be evaluated and compared, ensuring that it does not degrade significantly when exposed to new data from the same distribution as its training set.

This process involves dividing the dataset into a training set and a test set, training the model on the training set, and validating it on the test set. This procedure was repeated multiple times as per the Stratified K-folds cross-validation (Cross-Validation (CV)) method, which was selected for its effectiveness in handling imbalances in the target variable. Although the dataset used, consisting of 3000 images evenly distributed across four classes of vineyard diseases, appears balanced as detailed in section 3.5.1, the choice of Stratified K-folds anticipates future applications. In real-world situations, the distribution of vineyard disease images may not remain uniform across all classes, with some diseases likely being more prevalent than others. Adopting Stratified K-folds ensures that the model's performance is robust, offering/providing a realistic preparation for potential future scenarios where the data may be imbalanced.

In the context of Deep Learning (DL), implementing cross-validation presents difficulties because of the significant computational resources needed to train several models. Therefore, it is usual to set aside a portion of the data for hold-out validation rather than using k-Fold or other cross-validation techniques. The Keras library allows for a validation split or data to be defined specifically for this reason. The dataset is typically split into three parts: one for training, one for validation during training, and one for final testing of the model. This method ensures that deep learning models are tested for their ability to generalize, making them ready for real-world use despite the difficulties associated with cross-validation in these cases.

The following code snippet illustrates the implementation of cross-validation using Stratified K-Fold, illustrating how to prepare the dataset and how to train and validate the models in different folds. Each model has its own *createModel* function, where a model matching the number of classes is created for each fold. The code serves as a generic CV implementation framework that can be adapted to different models by customizing the model creation phase.

```
#Stratified K-Fold
skf = StratifiedKFold(n_splits=5, shuffle=True, random_state=1)

#Separate in cross-validation data and test data
cv_df, test_df = train_test_split(image_df, test_size=0.2,
stratify=image_df['Label'], random_state=1)

#Convert labels to categorical indices
cv_df['Label_encoded'] = LabelEncoder()
```

```
                              .fit_transform(cv_df['Label'])
num_classes = len(np.unique(cv_df['Label_encoded']))

# Main cross-validation loop
results = []

for train_index, val_index in
            skf.split(cv_df['Filepath'],cv_df['Label_encoded']):

    train_df = cv_df.iloc[train_index]
    val_df = cv_df.iloc[val_index]

    train_generator = create_gen(train_df, 'Label')
    val_generator = create_gen(val_df, 'Label')

    model = createModel(num_classes)

    history = model.fit(
        train_generator,
        epochs=50,
        validation_data=val_generator,
        verbose=1
    )
    results.append(history.history)
```

Listing 5.2: Cross-Valiadation implementation

5.1.2 Users evaluation

Following the models assessment, they are integrated into a final user-facing application, marking the beginning of the second phase of evaluation. This phase involves testing with end users who represent the potential real-world users of the application. The demographic for this test group is varied, it includes technicians as well as winegrowers who may interact with such applications in everyday scenarios.

The user evaluation is organized to measure the application's ease of use, functionality, and overall user satisfaction. Key aspects such as the user interface design, the intuitiveness of the navigation, and the clarity of the information presented are considered. Feedback is solicited through questionnaires and tasks, allowing users to provide insights into their experiences with the application.

The application performance is also evaluated in real-world conditions, exclusively through observation questionnaires and user feedback, focusing on user experience aspects.

The feedback collection method uses both quantitative and qualitative methods. Questions were created on Google Forms to facilitate data distribution and synthesis, they divided into sections corresponding to the tasks identified in table A.1. These tasks have been carefully selected based on their importance and relevance to the use cases that matter most to the success of the application. Some of these tasks are specific to features found only in SmartData platform, while others are common to both SmartData and AgriDash platforms.

Each section of the questionnaire refers to different aspects of the application, ensuring that feedback is focused and relevant to particular functionalities. As users engage with the tasks, they are asked to rate their experiences using a Likert scale, which provides a quantitative measure of their satisfaction.

The notes described in the table A.1 serve as a guide for the observer to capture the details of the user's interaction with the application. These notes will include any difficulties users encounter, the intuitiveness of the interface, and how well the tasks matched the users' expectations and real-world needs. Observations will be systematically recorded to identify any recurrent patterns or singular issues that could inform potential improvements. The results of this evaluation process will provide an overview of the project success and its areas for improvement.

5.2 Results

Following the evaluation of both the convolutional neural network (CNN) models and the end-user application, this section presents the results and their analysis. This section is divided into two phases: evaluation of the model and testing of the final application's usability by end users. These stages were meticulously designed to not only test the effectiveness of the CNN model in classifying diseases in vineyards, but also to evaluate real-world applicability and user satisfaction with the application of the developed application.

As discussed in the previous section, the evaluation process included subjecting the custom CNN, ResNet50 and MobileNetV2 models to testing on a test set of previously unseen vineyard images. This is done to evaluate their accuracy, precision, recall and F1-score, thus determining its effectiveness in real-life situations. This evaluation metrics, calculated based on model performance on the test set and through cross-validation methods, provide information about the robustness and reliability of each model.

At the same time, evaluating the usability of the application includes a series of user experience tests. These tests, designed around key tasks derived from the most relevant use cases allow us to collect valuable feedback from diverse groups of end-users, including technicians and winegrowers. The feedback obtained through Google Forms and organized by tasks described in table A.1 was analyzed to extract quantitative and qualitative data related to the application's functionality, ease of use, and overall user satisfaction.

This section aims to summarize the results of the two evaluation phases. It will detail the performance of each CNN model, comparing their strengths and weaknesses across different metrics. Additionally, it will consider user feedback, highlighting how the application meets the needs of its intended users and where improvements can be made.

5.2.1 Models

The evaluation results of the CNN models – custom CNN, ResNet50 and MobileNetV2 are divided into two parts: initial testing on a test set and further evaluation through cross-validation.

First, the performance of each model is examined on a test set composed of images not seen during training (see code in Listing 5.1). This step evaluates the models' ability to generalize to new data, a crucial factor for real-world applications.

Secondly, cross-validation is implemented to provide deeper insight into the reliability and consistency of models across different data splits (see code in Listing 5.2). This method helps to mitigate any bias and check the robustness of the models.

For both testing phases, metrics such as accuracy, precision, recall and F1 score were calculated to evaluate the models. These metrics are calculated directly from test set results offering a bigger understanding of each model's strengths and limitations in classifying vineyard diseases.

5.2.1.1 Test set results

Here, are presented the results of the evaluation of CNN models on the test set, with and without the application of cross-validation techniques. To ensure a fair comparison, all models were trained and evaluated under the same conditions, including the number of epochs as 50, optimizer settings as Adam optimizer with learning rate 0.0001, and a batch size of 32.

First, the performance of the models directly on the test set is discussed to evaluate their prediction ability on unseen data. Next, the results obtained through cross-validation are further explored, providing a clearer view of the consistency and reliability of each model in different subsets of data.

Without cross-validation

The custom CNN model achieved a Top-1 accuracy of 97.21%, demonstrating a high level of precision in correctly predicting the main class for each image.

The modified ResNet50, which was adapted for this specific task, attain an almost perfect Top-1 accuracy of 99.70%. A high Top-1 value indicates that the model is extremely reliable in classifying vineyard diseases. With a minimum value of 0.005 than the other models, is indicative of the model's precise predictions and its effectiveness for domain-specific tasks.

Similarly, the modified MobileNetV2 model gives great results with a Top-1 accuracy of 99.33%. Combined with a low test loss of 0.03, MobileNetV2's performance demonstrates its capabilities as a powerful model for applications where size and speed are as important as accuracy.

	CNN model	**modified ResNet50**	**modified MobileNetV2**
Test Loss	0.098	0.005	0.03
Top-1 Accuracy	97.21%	99.70%	99.33%

Table 5.1: Results on the test set without cross-validation.

These results, especially the high Top-1 accuracy score and minimal test loss, highlight the potential of these models to become reliable tools for disease detection in vineyards. When deciding which model to adopt for practical implementation, is necessary to consider additional factors such as its efficiency, ease of integration, and scalability. With great performance and low check loss, the modified ResNet50 emerges as the correct choice for model implementation in "Disease Classification"use case.

With cross-validation

For the custom CNN model, the cross-validation procedure showed a mean training accuracy of 99.97% and a mean validation accuracy of 98.21%. These high percentages suggest that the model has learned the necessary features to identify vineyard diseases with a high confidence. Although the average validation loss was slightly higher at 0.07 compared to the training loss at 0.002, the custom CNN model demonstrated strong classification accuracy, making it a candidate for practical applications.

The modified ResNet50 model shows greater results, achieving a perfect average training accuracy and an average validation accuracy of 99.84%. This near-perfect performance is further highlighted by an almost negligible average authentication loss of 0.0068. The consistency between training and validation results shows that the modifed ResNet50 model learned effectively to generalize the training data to new unseen images without overfitting.

Similar to the previous model, the modified MobileNetV2 model also performed very well, with perfect training accuracy and a mean validation accuracy of 99.44%. Although there is a small drop in validation accuracy compared to ResNet50, it still remains significantly high. A mean validation loss of 0.0209, while higher than ResNet50's, is still low enough to be considered good in most cases.

	CNN model	**modified ResNet50**	**modified MobileNetV2**
Test Loss	0.08	0.002	0.008
Top-1 Accuracy	98.04%	99.96%	99.75%

Table 5.2: Results on the test set with cross-validation.

When examining the results presented in Table 5.2, it is clear that each model has its advantages. Although the custom CNN model did not achieve validation accuracies as high as the modified ResNet50 or MobileNetV2, its strong Top-1 accuracy on the cross-validated test set demonstrate its competence. However, based on the cross-validation results, the modified ResNet50 model stands out for its balance between high accuracy and low loss, showing its superior performance and readiness for deployment in vineyard disease detection.

5.2.1.2 Metrics results

The comprehensive evaluation of our CNN models is illustrated by a set of critical performance metrics—accuracy, precision, recall (sensitivity), and the F1-Score—both with and without the application of cross-validation techniques. These metrics provide an in-depth look at the models' performance, extending beyond mere accuracy rates to gauge the quality of the predictions made.

Evaluation of the CNN models, carried out according to the code snippet provided in Listing 5.3, allows a detail analysis of each model's ability to accurately classify vineyard diseases. Using this code, critical performance metrics, including accuracy, precision, recall, and F1 score, are calculated for each model (as seen in Table 5.3).

```
#Evaluate the Model on the test set:
predictions = model.predict(test_images)
pred_labels = np.argmax(predictions, axis=1)

accuracy = accuracy_score(numerical_test_labels, pred_labels)
precision = precision_score(numerical_test_labels, pred_labels,
                            average='weighted')
recall = recall_score(numerical_test_labels, pred_labels,
                            average='weighted')
f1_score_result = f1_score(numerical_test_labels, pred_labels,
                            average='weighted')

print("Model Evaluation:")
print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1-Score:", f1_score_result)
```

Listing 5.3: Metrics evaluation on test set

Accuracy, the most direct of these metrics, showed that the custom CNN model achieved a high rate of correct predictions with 97.08%, while the modified ResNet50 and MobileNetV2 models demonstrated superior accuracy of 99.79% and 99.33%, respectively.

Precision reflects the proportion of true positive predictions against all positive calls made by the models, with all models showing a high precision rate, echoing the high quality of their predictive abilities.

Recall or sensitivity, which is vital in ensuring that no instance of disease is missed, was also high for all models, a crucial feature for potential deployment in critical agricultural settings where missing an instance of disease could have serious repercussions. The *F1-Score*, which synthesizes precision and recall, further highlighted the well-balanced nature of the models, with both the custom CNN and the modified models achieving F1-Scores that closely mirrored their high accuracy rates.

	CNN model	**modified ResNet50**	**modified MobileNetV2**
Accuracy	97.208%	99.79%	99.333%
Precision	97.288%	99.79%	99.339%
Recall	97.208%	99.79%	99.333%
F1-Score	97.228%	99.79%	99.333%

Table 5.3: Evaluation metrics for each model.

Choosing the ideal model to deploy requires a balance between accuracy and performance. The modified ResNet50 leads the way with outstanding accuracy, making it a top candidate for situations where accuracy is critical. However, the commendable accuracy of the MobileNetV2 model, combined with its computational efficiency, makes a case for its use where computational resources are limited or processing demands fast environments. Ultimately, the choice between the two models will be determined by the specific requirements of the application context, such as the need for real-time analytics or available computing power. One possible solution is to integrate the MobileNetV2 model locally into the AgriDash application, capitalizing on its strengths in real-time analytics and its suitability for environments with limited computational resources. In contrast, the ResNet50 model could/should be reserved for scenarios within the SmartData platform where accuracy is essential and computational resources are abundant, allowing slightly higher precision to be used without being limited.

5.2.2 Users

This section summarizes the feedback received from the expert technical agronomist after his interaction with the application, focusing on domain-specific issues that arose during use. These questions were prepared based on detailed notes and observations recorded while carrying out the tasks, as described in Table A.2 and the System Usability Scale (SUS) described in [Mar+15], see Table A.3 for the detailed questions. SUS scores, combined with qualitative feedback, serve to map areas for refinement in the application, both in the dataset and in the user interface.

The following tasks were highlighted for evaluation, each accompanied by respective feedback:

- **Satellite Layers Visualization Task** - From domain-specific questions, feedback indicated a clear preference for higher resolution satellite imagery, although Sentinel-2 data was considered sufficient for field workers and technicians. The user expressed the need for capabilities to set viewing dates, emphasizing that while current data is useful, access to historical data is essential for effective fieldwork. His insight also pointed out that this module takes more into account the needs of technicians than farmers, where the NDVI index is the most used. This suggests that the main users of this module will, in fact, be agronomic technicians.

- **Disease Classification Task** - The user noted the need for improvements to the dataset, such as including photos from the back of the plants, to better distinguish between similar diseases for example downy mildew and black rot. It also suggested adding a new classification layer to differentiate between diseases and nutritional needs, pointing to the potential for expanding the dataset to capture more disease patterns. Additionally, specific diseases such as downy mildew, powdery mildew, black rot and botrytis are crucial for inclusion. Contrary to what was said in the previous task, plant disease classification features are more aimed at farmers to help them correctly identify plant diseases and problems that are often difficult to detect with the naked eye.

- **Filter and Select Marker Task** - Although some diseases do not have direct treatments and only require notification, the application could also be extended to the detection of pests and potential nutritional deficiencies.

Overall feedback on the interface was positive, suggesting that although the algorithm and data set need improvement, the current configuration is still fit for its intended use. There was consensus that the application showed promise, with recommendations for refining its capabilities based on the comprehensive feedback collected.

6

Conclusion and Future Work

This chapter presents the conclusions of the work developed as part of this dissertation. Additionally, these findings allow for further analysis of the impact this work had on the contributions defined in Chapter 1. Finally, the chapter provides improvement ideas that may enhance the system's usability and performance.

6.1 Conclusion

The work developed throughout this dissertation aimed to fulfill the motivations outlined in Chapter 1 and at the same time make significant advances in the integration of advanced technologies into sustainable agricultural practices. Exploring Precision Agriculture (PA) techniques reveals potential insights into how IoT and machine learning could enhance farming efficiency and sustainability. The development of a robust API and a user-friendly interface, coupled with geo-locational mapping through GPS technology, has facilitated a deeper understanding of agronomic conditions. Furthermore, the interpretation of Sentinel-2 satellite data through advanced mathematical algorithms has provided valuable insights into soil health and vegetation vigor, thus aiding in informed decision-making processes.

The implementation of deep learning techniques for the detection of grapevine diseases has proven to be a important advancement. By developing models that leverage the capabilities of CNN, ResNet50 and MobileNetV2, it has been demonstrated that it is possible to achieve high accuracy in disease identification, which is crucial for timely interventions and outbreak prevention. The continuous learning model ensures that the system evolves and improves its diagnostic capabilities over time, based on data and images provided by the user.

The evaluation of these technologies, both from a technical and end-user point of view, has produced promising results. The models have demonstrated remarkable F1 accuracy, precision, and accuracy in classifying vineyard diseases, underlining their potential for real-world applications. Although the modified ResNet50 emerged as the most accurate, making it ideal for situations that require high precision, the efficiency of MobileNetV2 has proven to be beneficial for real-time analysis and situations where computational resources are low. The choice of model for implementation in a practical environment will depend significantly on the specific requirements of the application context, taking into account the need for accuracy versus the availability of computational resources. As such, the strategic approach involves incorporating ResNet50 into the SmartData framework for model processing locally, while adopting MobileNetV2 for the AgriDash application allows for remote model execution on mobile devices.

In turn, user feedback highlighted the practicality of the application, emphasizing its ease of use, functionality and the value it brings to technicians and winegrowers. The user evaluation phase provided critical insights into the real-world applicability of the developed application. The technical agronomist's feedback confirmed the application's potential to improve decision-making in vineyard management. The ability to review historical and current data, the preference for higher resolution images, and the call for a broader spectrum of disease and plant deficiency detection were among the highlighted areas for improvement. The interface's positive reception suggests a promising start, while suggestions for improvements offer a clear direction for refinement.

In conclusion, from a general perspective, it is safe to assume that the contributions initially outlined in Chapter 1 have been realized. This dissertation represents a significant contribution to the field of precision agriculture. By effectively marrying technology with traditional agricultural practices, it was possible to offer a model that not only increase the efficiency and sustainability of agriculture, but also open new paths for research and development in the agricultural sector. The positive feedback from technical evaluations and end-user testing highlights the real-world applicability and potential impact of this work on the agricultural community.

6.2 Future Work

Despite the satisfactory performance and feedback of the developed system, there is still room for improvement in some of its areas. The following list depicts a series of modifications that can improve overall system performance:

- **Enhanced Satellite Imagery Resolution:** While Sentinel-2 satellite imagery has been sufficient, acquiring higher-resolution images could significantly enhance the level of detail and accuracy in analyses. Efforts to access or purchase improved satellite imagery should be considered to augment the application's capabilities in monitoring and managing vineyard health.

- **Sensor Integration for Real-time Data Collection:** Incorporating sensors to collect real-time data on soil moisture, temperature and other environmental factors can provide more dynamic information to models, improving their accuracy and the granularity of the insights provided. This would also enable the development of predictive models that can predict disease outbreaks or adverse conditions before they occur.

- **Expanded User Testing:** Conducting more comprehensive user testing involving a broader demographic of end-users, including farmers, agronomists, and agricultural technicians, can yield deeper insights into the application's usability and areas for improvement. This feedback is crucial for refining the user interface, improving data visualization tools, and adapting the application to the varied needs of its users. To support this process, reference is made to the SUS Table A.3, which details the specific questions to be addressed to users during testing. These questions aim to evaluate performance, satisfaction and any difficulties encountered when using the application, ensuring a comprehensive analysis of the user experience.

- **Dataset Enrichment and Expansion:** Enhancing the dataset with images of diseases under varying conditions and stages of development, as well as including images from different angles, can improve the model's diagnostic capabilities. Expanding the dataset to include other crops and diseases would also increase the application's utility across the agricultural sector.

- **Integration with Agronomic Information:** Linking the application with existing agronomic information could provide users with a more comprehensive tool set. For instance, integrating detailed vineyard meta-information, such as grape variety and planting density, can enhance disease prediction models and offer more personalized management recommendations.

- **Development of a Pest and Nutrient Deficiency Module:** Extending the application's capabilities to include the detection of pests and the identification of nutrient deficiencies would provide farmers with a more holistic tool for crop management. This would involve the development of new models and the expansion of the existing dataset to include pest manifestations and symptoms of nutrient deficiencies.

Bibliography

[Abi+20] E. A. Abioye et al. "A review on monitoring and advanced control strategies for precision irrigation". In: *Computers and Electronics in Agriculture* 173 (2020), p. 105441. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2020.105441. URL: https://www.sciencedirect.com/science/article/pii/S0168169919314826 (cit. on p. 8).

[Age] E. S. Agency. *SNAP*. URL: http://step.esa.int/main/download/snap-download/ (cit. on p. 45).

[Agr13] A. D. A. S. for Agri-Food Value Chain. *Agrivi*. 2013. URL: https://www.agrivi.com/ (cit. on pp. 27, 28).

[Ali+22] K. Alibabaei et al. "A Review of the Challenges of Using Deep Learning Algorithms to Support Decision-Making in Agricultural Activities". In: *Remote Sensing* 14.3 (2022). ISSN: 2072-4292. DOI: 10.3390/rs14030638. URL: https://www.mdpi.com/2072-4292/14/3/638 (cit. on p. 14).

[Ana24a] E. D. Analytics. *EOS Data Analytics: Understanding NDMI*. 2024. URL: https://eos.com/make-an-analysis/ndmi/ (cit. on p. 49).

[Ana24b] E. D. Analytics. *EOS Data Analytics: Understanding NDVI*. 2024. URL: https://eos.com/make-an-analysis/ndvi/ (cit. on pp. 47, 48).

[Ana24c] E. D. Analytics. *Normalized Difference Water Index (NDWI)*. 2024. URL: https://eos.com/make-an-analysis/ndwi/ (cit. on p. 53).

[AVI21] AVIPE. *Doenças*. 2021. URL: https://www.avipe.pt/doen%C3%A7as (cit. on pp. 66, 73).

[Bai+22] D. Bai et al. "Impact of climate change on agricultural productivity: a combination of spatial Durbin model and entropy approaches". In: (2022-10). DOI: 10.1108/ijccsm-02-2022-0016 (cit. on p. 55).

[Ben+21] A. Benos L.and Tagarakis et al. "Machine Learning in Agriculture: A Comprehensive Updated Review". In: *Sensors* 21 (2021), p. 3758. DOI: https://doi.org/10.3390/s21113758 (cit. on p. 10).

[BH21] S. A. Bhat and N.-F. Huang. "Big Data and AI Revolution in Precision Agriculture: Survey and Challenges". In: *IEEE Access* 9 (2021), pp. 110209–110222. DOI: 10.1109/ACCESS.2021.3102227 (cit. on pp. 22, 23).

[Bus11] Bushel. *FarmLogs*. 2011. URL: https://farmlogs.com/ (cit. on p. 28).

[Car] P. Carvalho. *Dynamics of rural low density spaces in Portugal*. DOI: https://doi.org/10.4000/mediterranee.10516. URL: http://journals.openedition.org/mediterranee/10516 (visited on 2018-11-28) (cit. on p. 3).

[Com] E. Commission. *Data and Information Access Services*. URL: https://www.copernicus.eu/en/access-data/dias (cit. on p. 45).

[Con22] I. S. W. Congress. *IOT TRANSFORMING THE FUTURE OF AGRICULTURE*. 2022. URL: https://www.iotsworldcongress.com/iot-transforming-the-future-of-agriculture/ (cit. on p. 10).

[Cra+22] A. Cravero et al. "Challenges to Use Machine Learning in Agricultural Big Data: A Systematic Literature Review". In: *Agronomy* 12.3 (2022). ISSN: 2073-4395. DOI: 10.3390/agronomy12030748. URL: https://www.mdpi.com/2073-4395/12/3/748 (cit. on pp. 21–23).

[Cro03] V. Crossing. *Weather Data & API. Global Forecast & History Data*. 2003. URL: https://www.visualcrossing.com/ (cit. on pp. 54, 56).

[DBF22] T. Domingues, T. Brandão, and J. C. Ferreira. "Machine Learning for Detection and Prediction of Crop Diseases and Pests: A Comprehensive Survey". In: *Agriculture* 12.9 (2022). ISSN: 2077-0472. DOI: 10.3390/agriculture12091350. URL: https://www.mdpi.com/2077-0472/12/9/1350 (cit. on p. 11).

[ED10] E. Earon and G. D'Eleuterio. *PrecisionHawk*. 2010. URL: https://www.precisionhawk.com/ (cit. on p. 27).

[Eli+18] O. Elijah et al. "An Overview of Internet of Things (IoT) and Data Analytics in Agriculture: Benefits and Challenges". In: *IEEE Internet of Things Journal* 5.5 (2018), pp. 3758–3773. DOI: 10.1109/JIOT.2018.2844296 (cit. on p. 10).

[Els+18] R. Elshawi et al. "Big Data Systems Meet Machine Learning Challenges: Towards Big Data Science as a Service". In: *Big Data Research* 14 (2018), pp. 1–11. ISSN: 2214-5796. DOI: https://doi.org/10.1016/j.bdr.2018.04.004. URL: https://www.sciencedirect.com/science/article/pii/S2214579617303957 (cit. on p. 21).

[EG03] R. E. Evenson and D. Gollin. "Assessing the Impact of the Green Revolution, 1960 to 2000". In: *Science* 300.5620 (2003), pp. 758–762. DOI: 10.1126/science.1078710. URL: https://www.science.org/doi/abs/10.1126/science.1078710 (cit. on p. 1).

[Fou+15] S. Fountas et al. "Farm management information systems: Current situation and future perspectives". In: *Computers and Electronics in Agriculture* 115 (2015), pp. 40–50. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2015.05.011. URL: https://www.sciencedirect.com/science/article/pii/S0168169915001337 (cit. on p. 3).

[FI19] M. Franzese and A. Iuliano. "Hidden markov models". In: (2019), pp. 1133–1141. DOI: https://doi.org/10.1016/B978-0-12-809633-8.20488-3. URL: https://www.sciencedirect.com/science/article/pii/B9780128096338204883 (cit. on p. 14).

[Fro09] J. Froikin-Gordon. *AgSquared*. 2009. URL: https://www.agsquared.com/ (cit. on p. 28).

[Gad+20] K. K. Gadiraju et al. "Multimodal Deep Learning Based Crop Classification Using Multispectral and Multitemporal Satellite Imagery". In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '20. Virtual Event, CA, USA: Association for Computing Machinery, 2020, pp. 3234–3242. ISBN: 9781450379984. DOI: 10.1145/3394486.3403375. URL: https://doi.org/10.1145/3394486.3403375 (cit. on p. 14).

[Gar+21] S. Garg et al. "Towards a Multimodal System for Precision Agriculture using IoT and Machine Learning". In: (2021-07). DOI: 10.1109/ICCCNT51525.2021.9579646 (cit. on p. 10).

[GT06] S. O. Geoffrey E Hinton and Y.-W. Teh. "A fast learning algorithm for deep belief nets". In: *Neural computation 18* 7 (2006), pp. 1527–1554. URL: https://www.cs.toronto.edu/~hinton/absps/fastnc.pdf (cit. on p. 12).

[God+10] H. C. J. Godfray et al. "Food Security: The Challenge of Feeding 9 Billion People". In: *Science* 327.5967 (2010), pp. 812–818. DOI: 10.1126/science.1185383. URL: https://www.science.org/doi/abs/10.1126/science.1185383 (cit. on p. 1).

[gov21] U. government. *GPS applications to agriculture*. 2021. URL: https://www.gps.gov/applications/agriculture/ (cit. on p. 7).

[GWW19] W. Guo, J. Wang, and S. Wang. "Deep Multimodal Representation Learning: A Survey". In: *IEEE Access* 7 (2019), pp. 63373–63394. DOI: 10.1109/ACCESS.2019.2916887 (cit. on p. 12).

[Han23] HandWiki. *Data visualization*. 2023. URL: https://handwiki.org/wiki/Data_visualization#Human_perception/cognition_and_data_visualization (cit. on pp. 24, 26).

[HS21] D. L. Hickman and S. J. Shepperd. "Image fusion systems for surveillance applications: design options and constraints for a tri-band camera". In: *Infrared Technology and Applications XLVII*. Ed. by B. F. Andresen, G. F. Fulop, and L. Zheng. Vol. 11741. International Society for Optics and Photonics. SPIE, 2021, p. 117411L. DOI: 10.1117/12.2584984. URL: https://doi.org/10.1117/12.2584984 (cit. on p. 20).

[Hou+17] X. Hou et al. "Deep Feature Consistent Variational Autoencoder". In: (2017), pp. 1133–1141. DOI: 10.1109/WACV.2017.131 (cit. on p. 12).

[HPS20] S. Huang, A. Pareek, and S. e. a. Seyyedi. "Fusion of medical imaging and electronic health records using deep learning: a systematic review and implementation guidelines." In: *npj Digit. Med.* 3 (2020), p. 136. DOI: https://doi.org/10.1038/s41746-020-00341-z (cit. on pp. 18–20).

[Hub16] S. Hub. *Cloud API for Satellite Imagery*. 2016. URL: https://www.sentinel-hub.com/ (cit. on p. 45).

[Ian+14] M. M. Ian Goodfellow Jean Pouget-Abadie et al. "Generative adversarial nets". In: *Advances in neural information processing systems* 27 (2014), pp. 2672–2680. URL: https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf (cit. on p. 14).

[Jác21] L.-R. Jácome-Galarza. ""Crop yield prediction utilizing multimodal deep learning"". In: *2021 16th Iberian Conference on Information Systems and Technologies (CISTI)*. 2021, pp. 1–6. DOI: 10.23919/CISTI52073.2021.9476420 (cit. on p. 14).

[JF16] R. G. Joseph Redmon Santosh Divvala and A. Farhadi. "You only look once: Unified, real-time object detection". In: *IEEE conference on computer vision and pattern recognition*. 2016, pp. 779–788. URL: https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Redmon_You_Only_Look_CVPR_2016_paper.pdf (cit. on p. 13).

[KKP17] A. Kamilaris, A. Kartakoullis, and F. X. Prenafeta-Boldú. "A review on the practice of big data analysis in agriculture". In: *Computers and Electronics in Agriculture* 143 (2017), pp. 23–37. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2017.09.037. URL: https://www.sciencedirect.com/science/article/pii/S0168169917301230 (cit. on pp. 21, 22).

[KKK21] H. Kaur, D. Koundal, and V. Kadyan. "Image Fusion Techniques: A Survey." In: *Arch Computat Methods Eng* 28 (2021), pp. 4425–4447. DOI: https://doi.org/10.1007/s11831-021-09540-7 (cit. on pp. 17, 19, 20).

[Kum+19] L. Kummar et al. "Data Visualization for Smart Farming Using Mobile Application". In: 19 (2019-12), pp. 1–7. URL: http://paper.ijcsns.org/07_book/201911/20191101.pdf (cit. on p. 24).

[Las+21] R. Lassoued et al. "Expert Insights on the Impacts of, and Potential for, Agricultural Big Data". In: *Sustainability* 13.5 (2021). ISSN: 2071-1050. DOI: 10.3390/su13052521. URL: https://www.mdpi.com/2071-1050/13/5/2521 (cit. on pp. 23, 24).

[LeC+89] Y. LeCun et al. "Backpropagation Applied to Handwritten Zip Code Recognition". In: *Neural Computation* 1.4 (1989), pp. 541–551. DOI: 10.1162/neco.1989.1.4.541 (cit. on p. 13).

[Li+21] D. Li et al. "Recent advances in image fusion technology in agriculture". In: *Computers and Electronics in Agriculture* 191 (2021), p. 106491. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2021.106491. URL: https://www.sciencedirect.com/science/article/pii/S0168169921005081 (cit. on p. 20).

[Lio+21] A. Liopa-Tsakalidi et al. "A NB-IoT based platform for smart irrigation in vineyard". In: *2021 10th International Conference on Modern Circuits and Systems Technologies (MOCAST)*. 2021, pp. 1–4. DOI: 10.1109/MOCAST52088.2021.9493381 (cit. on p. 10).

[Lou21] J. M. Lourenço. *The NOVAthesis LaTeX Template User's Manual*. NOVA University Lisbon. 2021. URL: https://github.com/joaomlourenco/novathesis/raw/main/template.pdf (cit. on p. ii).

[M97] D. J. M. *Guns Germs and Steel : The Fates of Human Societies*. First ed. W.W. Norton & Company, 1997. ISBN: 9780393038910. DOI: 10.1017/S0022050700022178 (cit. on p. 1).

[Mad+24] M. Madeira et al. "AI-powered Solution for Plant Disease Detection in Viticulture". In: *Procedia Computer Science* 238 (2024). The 15th International Conference on Ambient Systems, Networks and Technologies Networks (ANT) / The 7th International Conference on Emerging Data and Industry 4.0 (EDI40), April 23-25, 2024, Hasselt University, Belgium, pp. 468–475. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2024.06.049. URL: https://www.sciencedirect.com/science/article/pii/S1877050924012857 (cit. on p. 4).

[Mad+22] R. N. Madeira et al. "Towards Digital Twins for Multi-Sensor Land and Plant Monitoring". In: *Procedia Computer Science* 210 (2022). The 13th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EU-SPN) / The 12th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2022) / Affiliated Workshops, pp. 45–52. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2022.10.118. URL: https://www.sciencedirect.com/science/article/pii/S1877050922015745 (cit. on p. 3).

[Man23] R. Mandal. *Augmented Grapevine Disease Dataset*. 2023. URL: https://www.kaggle.com/datasets/rm1000/augmented-grape-disease-detection-dataset (cit. on p. 38).

[MY21] C. Manliguez and J. Y. Chiang. "Multimodal Deep Learning and Visible-Light and Hyperspectral Imaging for Fruit Maturity Estimation". In: *Sensors* 21 (2021-02), p. 1288. DOI: 10.3390/s21041288 (cit. on p. 14).

[Map10] Mapbox. *Mapbox Documentation*. 2010. URL: https://docs.mapbox.com/#maps (cit. on pp. 31, 62).

[Mar+15] A. I. Martins et al. "European Portuguese Validation of the System Usability Scale (SUS)". In: *Procedia Computer Science* 67 (2015). Proceedings of the 6th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion, pp. 293–300. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2015.09.273. URL: https://www.sciencedirect.com/science/article/pii/S1877050915031191 (cit. on p. 83).

[Mas+17] S. Masood et al. "Image Fusion Methods: A Survey". In: *Journal of Engineering Science and Technology Review* 10 (2017-12), pp. 186–195. DOI: 10.25103/jestr.106.24 (cit. on pp. 16, 17).

[McB+05] A. McBratney et al. "Future Directions of Precision Agriculture". In: *Precision Agric* 6 (2005), pp. 7–23. DOI: https://doi.org/10.1007/s11119-005-0681-8 (cit. on p. 6).

[ML21] B. R. Meena and D. B. L. "Precision Farming; their tools and techniques". In: *Just Agriculture* 2 (2021-10), pp. 2692–2697. ISSN: 2582-8223. URL: https://justagriculture.in/files/newsletter/2021/september/49.%20Precision%20Farming%20their%20tools%20and%20techniques.pdf (cit. on p. 8).

[Mis+22] N. N. Misra et al. "IoT, Big Data, and Artificial Intelligence in Agriculture and Food Industry". In: *IEEE Internet of Things Journal* 9.9 (2022), pp. 6305–6324. DOI: 10.1109/JIOT.2020.2998584 (cit. on p. 20).

[MH20] A. A. E. Mohd Saiful Azimi Mahmud Mohamad Shukri Zainal Abidin and H. S. Hasan. "Robotics and Automation in Agriculture: Present and Future Applications". In: vol. 4. UTM Skudai, Johor, Malaysia: School of Electrical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, 2020, pp. 130–140. URL: https://www.researchgate.net/publication/340397309_Robotics_and_Automation_in_Agriculture_Present_and_Future_Applications (cit. on p. 8).

[NAS00] NASA. *Measuring Vegetation (NDVI & EVI)*. 0200. URL: https://earthobservatory.nasa.gov/features/MeasuringVegetation/measuring_vegetation_4.php (cit. on p. 9).

[NET24] NETAFIM. *Definition of Precision Irrigation*. 2024. URL: https://www.netafimindia.com/precision-Irrigation/ (cit. on p. 8).

[Ouh+21] M. Ouhami et al. "Computer Vision, IoT and Data Fusion for Crop Disease Detection Using Machine Learning: A Survey and Ongoing Research". In: *Remote Sensing* 13.13 (2021). ISSN: 2072-4292. DOI: 10.3390/rs13132486. URL: https://www.mdpi.com/2072-4292/13/13/2486 (cit. on p. 11).

[RW14] C. Rudin and K. Wagstaff. "Machine learning for science and society". In: *Mach Learn* 95 (2014), pp. 1–9. DOI: https://doi.org/10.1007/s10994-013-5425-9 (cit. on p. 10).

[SC10] A. S. S. Li and B. A. Chin. "Sensors for Agriculture and the Food Industry". In: *The Electrochemical Society Interface* 19.4 (2010), pp. 41–46. DOI: 10.1149/2.f05104if. URL: https://doi.org/10.1149/2.f05104if (cit. on p. 10).

[Sah+19] B. Sahu et al. "Tools of precision agriculture: A review". In: *International Journal of Chemical Studies* 7 (2019-12), pp. 2692–2697. URL: https://www.researchgate.net/publication/338502162_Tools_of_precision_agriculture_A_review (cit. on p. 8).

[SH09] R. Salakhutdinov and G. Hinton. "Deep boltzmann machines". In: *In Artificial intelligence and statistics* (2009), pp. 448–455. URL: https://www.cs.toronto.edu/~fritz/absps/dbm.pdf (cit. on p. 12).

[Sar+19] M. Sarker et al. "Promoting digital agriculture through Big Data for sustainable farm management". In: *Int. J. Innov. Appl. Stud.* 25 (2019), pp. 1235–1240. URL: https://tapipedia.org/content/promoting-digital-agriculture-through-big-data-sustainable-farm-management (cit. on p. 21).

[SOA19] I. SASSI, S. OUAFTOUH, and S. ANTER. "Adaptation of Classical Machine Learning Algorithms to Big Data Context: Problems and Challenges : Case Study: Hidden Markov Models Under Spark". In: *2019 1st International Conference on Smart Systems and Data Science (ICSSD)*. 2019, pp. 1–7. DOI: 10.1109/ICSSD47982.2019.9002857 (cit. on p. 21).

[Sch17] K. Schwab. *The Fourth Industrial Revolution*. USA: Crown Publishing Group, 2017. ISBN: 1524758868. DOI: https://dl.acm.org/doi/10.5555/3137529 (cit. on p. 1).

[Ser14] Serco. *User Guide Open Access Hub*. 2014. URL: https://scihub.copernicus.eu/userguide/ (cit. on p. 44).

[Ser23] Serco. *Open Access Hub*. 2023. URL: https://scihub.copernicus.eu/ (cit. on p. 44).

[Smo86] P. Smolensky. "Information processing in dynamical systems: Foundations of harmony theory". In: *Technical Report* (1986). URL: https://stanford.edu/~jlmcc/papers/PDP/Volume%201/Chap6_PDP86.pdf (cit. on p. 12).

[Spr07] Springer. *Precision Agriculture definition*. 2007. URL: https://www.springer.com/journal/11119/updates/17240272 (cit. on p. 6).

[Ste+20] S. Stevanoska et al. "IoT–Based System for Real-Time Monitoring and Insect Detection in Vineyards". In: *Proceedings of the 18th ACM Symposium on Mobility Management and Wireless Access*. MobiWac '20. Alicante, Spain: Association for Computing Machinery, 2020, pp. 133–136. ISBN: 9781450381192. DOI: 10.1145/3416012.3424634. URL: https://doi.org/10.1145/3416012.3424634 (cit. on p. 10).

[Sum+21] J. Summaira et al. *Recent Advances and Trends in Multimodal Deep Learning: A Review*. 2021. DOI: 10.48550/ARXIV.2105.11087. URL: https://arxiv.org/abs/2105.11087 (cit. on p. 12).

[Sut16] S. Suthaharan. "Support Vector Machine". In: *Machine Learning Models and Algorithms for Big Data Classification* (2016), pp. 207–235. DOI: https://doi.org/10.1007/978-1-4899-7641-3_9 (cit. on p. 13).

[Tea02] Q. D. Team. *QGIS*. 2002. URL: https://qgis.org/en/site/ (cit. on p. 45).

[Tia16] F. Tian. "An agri-food supply chain traceability system for China based on RFID & blockchain technology". In: *2016 13th International Conference on Service Systems and Service Management (ICSSSM)*. 2016, pp. 1–6. DOI: 10.1109/ICSSSM.2016.7538424 (cit. on p. 3).

[UN] UN. *Population*. URL: https://www.un.org/en/global-issues/population#:~:text=Our%20growing%20population&text=The%20world's%20population%20is%20expected,billion%20in%20the%20mid%2D2080s. (cit. on p. 2).

[Vit84] A. de Viticultores do Concelho de Palmela. *Apresentação AVIPE*. 1984. URL: https://www.avipe.pt/ (cit. on p. 3).

[Wee+18] A. Weersink et al. "Opportunities and Challenges for Big Data in Agricultural and Environmental Analysis". In: *Annual Review of Resource Economics* 10.1 (2018), pp. 19–37. DOI: 10.1146/annurev-resource-100516-053654. URL: https://doi.org/10.1146/annurev-resource-100516-053654 (cit. on pp. 22, 23).

[WJD20] M. Weiss, F. Jacob, and G. Duveiller. "Remote sensing for agricultural applications: A meta-review". In: *Remote Sensing of Environment* 236 (2020), p. 111402. ISSN: 0034-4257. DOI: https://doi.org/10.1016/j.rse.2019.111402. URL: https://www.sciencedirect.com/science/article/pii/S0034425719304213 (cit. on p. 7).

[WCC14] A. Wezel, M. Casagrande, and F. e. a. Celette. "Agroecological practices for sustainable agriculture. A review". In: *Agron. Sustain. Dev.* 34 (2014), pp. 1–20. DOI: https://doi.org/10.1007/s13593-013-0180-7 (cit. on p. 8).

[WTA21] E. White, J. Thomasson, and B. e. a. Auvermann. “Report from the conference, ’identifying obstacles to applying big data in agriculture’”. In: *Precision Agric* 22 (2021), pp. 306–315. DOI: https://doi.org/10.1007/s11119-020-09738-y (cit. on p. 23).

[Wig] I. Wigmore. *precision agriculture*. URL: https://techtarget.com/whatis/definition/precision-agriculture-precision-farming (cit. on p. 2).

[Yan11] D. Yan-e. “Design of Intelligent Agriculture Management Information System Based on IoT”. In: *Fourth International Conference on Intelligent Computation Technology and Automation* 1 (2011), pp. 1045–1049. DOI: 10.1109/ICICTA.2011.262 (cit. on p. 9).

[Yua+22] Y. Yuan et al. “Advanced agricultural disease image recognition technologies: A review”. In: *Information Processing in Agriculture* 9.1 (2022), pp. 48–59. ISSN: 2214-3173. DOI: https://doi.org/10.1016/j.inpa.2021.01.003. URL: https://www.sciencedirect.com/science/article/pii/S2214317321000032 (cit. on p. 14).

[ZK12] C. Zhang and J. Kovacs. “The application of small unmanned aerial systems for precision agriculture: a review.” In: *Precision Agric* 13 (2012), pp. 693–712. DOI: 10.1007/s11119-012-9274-5. URL: https://doi.org/10.1007/s11119-012-9274-5 (cit. on p. 1).

[Zha+17] L. Zhang et al. “From Macro to Micro: Autonomous Multiscale Image Fusion for Robotic Surgery”. In: *IEEE Robotics & Automation Magazine* 24.2 (2017), pp. 63–72. DOI: 10.1109/MRA.2017.2680543 (cit. on p. 20).

[Zha+19] Y. Zhang et al. “Exploration of Deep Learning-based Multimodal Fusion for Semantic Road Scene Segmentation”. In: *VISAPP 2019 14th International Conference on Computer Vision Theory and Applications*. Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. Prague, Czech Republic, 2019. DOI: 10.5220/0007360403360343. URL: https://hal-univ-bourgogne.archives-ouvertes.fr/hal-02060222 (cit. on pp. 18, 20).

A

API Evaluation Materials

Tasks	Steps to follow
Map interaction	1. After logging in, access the "Satellite" window under "Data"; 2. Zoom in and zoom out; 3. Double "click" on the screen; 4. Select polygons/plots; 5. Drag map; 6. Tilt satellite view (3D);
Create polygons	1. Select polygon creation icon; 2. Define on map at least three points fully connected; 3. Drag polygon; 4. Edit polygon border; 5. Select polygon; 6. Return to step 1 for multiple polygon creation;
Satellite Layers Visualization	1. Select polygon; 2. Choosing a satellite index to view: 2.1. Successfully: 2.1.1. Interact with map; 2.2. Unsuccessful: 2.2.1. Read warning; 2.2.2. Increase viewing days; 3. Return to step 2 or 1;
Disease Classification	1. After logging in, access the "Disease Classification" window under "Data"; 2. Select the "Submit image" button; 3. Choose image to classify; 4. Select the "Run classification algorithm" button;
Filter and select markers	1. Access the "Satellite" window in "Data"; 2. Select button Marker symbol; 3. Choose a filter for viewing; 4. Select a white/normal marker; 5. Select a disease marker (non-white): 5.1. Select the "Treaty" button;
View weather data	1. Select or create a polygon; 2. Select which weather type to view; 3. Select which weather feature to view; 4. Select which value to view (max, min, avg); 5. Zoom in and zoom out; 6. Return to step 1, 2 or 3;

Table A.1: End-users tasks to be evaluated

Tasks	Notes
Map interaction	- Check if zoom min and max are adequate. - Check if interactions are adequate. - Check if polygons selection color is noticeable.
Create polygons	- Only for SmartData. - Check if polygon border color are noticeable. - Check dificulty on editing polygons.
Satellite Layers Visualization	- Check if satellite indices are sufficient. - Check if index legend is perceptible. - Check if the colors of the satellite layers are perceptible. - The information in the error window is adequate. - Check if the buttons to adjust the viewing days are appropriate.
Disease Classification	- Check if the image provided remains legible. - Check if the result text is adequate and sufficient. - Check if the image provided as an example of disease remains legible and perceptible.
Filter and select markers	- Check if the filter is appropriate and intuitive. - Check if the marker colors are legible. - Check if the information contained in the marker pop-up is sufficient and legible.
View weather data	- Only for SmartData. - Check if graph is clear and intuitive. - Check location weather data accuracy.

Table A.2: Tasks notes

SUS questions	Corresponding questions in Portuguese
I think that I would like to use this system frequently.	Acho que gostaria de utilizar este produto com frequência.
I found the system unnecessarily complex.	Considerei o produto mais complexo do que necessário.
I thought the system was easy to use.	Achei o produto fácil de utilizar.
I think that I would need the support of a technical person to be able to use this system.	Acho que necessitaria de ajuda de um técnico para conseguir utilizar este produto.
I found the various functions in this system were well integrated.	Considerei que as várias funcionalidades deste produto estavam bem integradas.
I thought there was too much inconsistency in this system.	Achei que este produto tinha muitas inconsistências.
I would imagine that most people would learn to use this system very quickly.	Suponho que a maioria das pessoas aprenderia a utilizar rapidamente este produto.
I found the system very cumbersome to use.	Considerei o produto muito complicado de utilizar.
I felt very confident using the system.	Senti-me muito confiante a utilizar este produto.
I needed to learn a lot of things before I could get going with this system.	Tive que aprender muito antes de conseguir lidar com este produto.

Table A.3: SUS questions and corresponding translation in Portuguese