Ana Beatriz de Tróia Salvado

Licenciado em Ciências da Engenharia Electrotécnica e de Computadores

Aerial Semantic Mapping for Precision Agriculture using Multispectral Imagery

Dissertação para obtenção do Grau de Mestre em

Engenharia Electrotécnica e de Computadores

Orientador: Prof. Dr. José António Barata de Oliveira, Prof. Associado, Universidade Nova de Lisboa

Co-orientador: Mestre Ricardo André Martins Mendonça, Investigador, Universidade Nova de Lisboa

Júri

Presidente: Prof. Dr. Luís Augusto Bica Gomes de Oliveira, FCT-UNL
Arguente: Prof. Dr. José Manuel Matos Ribeiro da Fonseca, FCT-UNL
Vogal: Prof. Dr. José António Barata de Oliveira, FCT-UNL

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Setembro, 2018

Aerial Semantic Mapping for Precision Agriculture using Multispectral Imagery

To my beloved family...

Acknowledgements

"Unity is strength... when there is teamwork and collaboration, wonderful things can be achieved." - Mattie Stepanek

Sharing moments, in a life full of daily routines, hard-work and commitment becomes true gold when it comes to find the right balance in life. Working together is the unity of growing with new ideas, solutions and knowledge to reach success. Family and Friendship is the magical formula to turn everything into happiness, laugh and crazy adventures, shaping who we are and the way we live. For this reason, I would like to leave a special "Thank You" to each person who crossed my path during this wonderful journey of learning.

I would like to thank Faculdade de Ciências e Tecnologias from Universidade Nova de Lisboa, for being my second home for the last five years. A peaceful place full of so many good people and which location allowed quick getaways to the coast with a wonderful view over the sea.

A special thank you to professor José Barata for being such an enthusiastic and welcoming person. Motivating us to dive into a wide range of opportunities and new projects full of innovation and technology. Thank you very much for letting me be part of RICS group, a team full of bright minds and personalities who, not only shared their lab with me, but also for the companionship, joy and playing moments, in a place where it is possible to work in a valuable environment, feeling the achievements of each other and progressing as a group.

Thank you Ricardo Mendonça, not only for the support and advice during all this working project but also for the leisure moments you provided everyday in that room, sharing your playlists in the RICS lab, by making it into a lighter environment with your great taste for background music. It was also a pleasure to share this lab with Francisco Marques, André Lourenço, Eduardo Pinto, Carlos Simões, Manuel Silva, Pedro Prates, João Lopes and Luís do Ó, who were always there to share their support, experience and guidance, to turn this project idea into reality.

A special thank you to my fellow colleagues and best friends on the board Carolina, David, João Morgado, Pedro, Alex and Bruno, for all the true friendship and wonderful moments shared, since the trips to Forum, the long queues and sudden change of plans about the place to have lunch, all the music: singing moments, playing the piano, the guitar

or David's drums. For all the evenings and projects that fomented the connection we have today. And also to my friends and companions who also left their footprint throughout this academic journey with me: Gisela, Prego, Sofia Pereira, Rita, Rúben, Rodrigo, Estevam, Andreia, Catarina and all the NEEC members.

Ponto Zero, a second family that I have the pleasure to be part of, for so many years: a team, a place, a gym, a group; sharing together not only the essence of gymnastics, but also the amazing moments of creativity, laugh and mutual help. Those goosebumps and nervous feeling before getting on stage, exhibitions and all the travels representing our country abroad. A huge thanks to my coaches João Martins and Pascoal Furtado for being in my life with such strong personalities, from whom I learnt so much and grew as a person.

Last but not least, thank you so much to my family and close friends, to my grandparents and godparents who always believed in me to achieve so much. But specially to my wonderful Mum who is always there for the best conversations, advice and contagious smile; to my dad not only for being a man always ready to share his wisdom with his daughters but also for his valuable serenity solving problems, playful mind and jokes; A huge "Thank You!" to my big sis Ana Rita, that despite all the challenges we go through, is always there to cheer me up, to give me such support whatever it takes and to give me all the strength to follow my dreams; and to João Pedro Carvalho, whose smile brightens up my life everyday, for all his love and positivism, believing that everything is possible.

Abstract

Nowadays constant technological evolution cover several necessities and daily tasks in our society. In particular, drones usage, given its wide vision to capture the terrain surface images, allows to collect large amounts of information with high efficiency, performance and accuracy.

This master dissertation's main purpose is the analysis, classification and respective mapping of different terrain types and characteristics, using multispectral imagery.

Solar radiation flow reflected on the surface is captured by the used multispectral camera's different lenses (RedEdge-M, created by Micasense). Each one of these five lenses is able to capture different colour spectrums (i.e. Blue, Green, Red, Near-Infrared and RedEdge). It is possible to analyse the various spectrum indices from the collected imagery, according to the fusion of different combinations between coloured bands (e.g. NDVI, ENDVI, RDVI...).

This project engages a ROS (Robot Operating System) framework development, capable of correcting different captured imagery and, hence, calculating the implemented spectral indices. Several parametrizations of terrain analysis were carried throughout the project, and this information was represented in semantic maps by layers (e.g. vegetation, water, soil, rocks).

The obtained experimental results were validated in the scope of several projects incorporated in PDR2020, with success rates between 70% and 90%.

This framework can have multiple technical applications, not only in Precision Agriculture, but also in vehicles autonomous navigation and multi-robot cooperation.

Keywords: Precision Agriculture, Layered Map, Semantic Map, Imagery Stitching, Unmanned Aerial Vehicle (UAV), Multispectral Imagery, Rededge-M, Micasense, 2D Perception.

Resumo

A constante evolução tecnológica cobre, hoje em dia, diversas necessidades e tarefas diárias da nossa sociedade. Particularmente, a inserção de drones, dada a sua ampla visão para captar imagens sobre a superfície do terreno, permite a recolha de grandes quantidades de informação com grande eficiência, desempenho e precisão.

A presente dissertação de mestrado tem como principal objetivo a análise, classificação e respetivo mapeamento de diferentes tipos e características de terrenos a partir de imagens multiespectrais.

O fluxo de radiação solar refletida pela superfície é captado pelas diferentes lentes da câmara multiespectral utilizada (RedEdge-M, produzida pela Micasense). Cada uma das cinco lentes é capaz de captar diferentes espectros de cor (i.e. Blue, Green, Red, Near-Infrared e RedEdge). A partir das imagens recolhidas é possível analisarem-se diversos índices espectrais consoante a fusão de distintas combinações entre as bandas de cor (e.g. NDVI, ENDVI, RDVI...).

Este projeto passa pelo desenvolvimento de uma nova framework em ROS (Robot Operating System) capaz de corrigir as diferentes imagens captadas e consequentemente calcular os índices espectrais implementados. Foram realizadas diversas parametrizações ao longo do projeto para a análise de terrenos, cuja informação é representada em mapas semânticos por camadas (e.g. vegetação, água, solo, rochas).

Os resultados experimentais obtidos foram validados no âmbito de vários projetos inseridos no PDR2020, com taxas de sucesso compreendidas entre 70% e 90%.

Esta framework poderá ter diversas aplicações técnicas não só na Agricultura de Precisão, como também para a navegação autónoma de veículos e ainda na cooperação multi-robot.

Palavras-chave: Agricultura de Precisão, Mapa de Camadas, Mapa Semântico, Junção Panorâmica de Imagens, Veículo Aéreo Não Tripulado, Imagens Multiespectrais, Rededge-M, Micasense, Percepção 2D.

Contents

List of Figures

List of Tables

Acronyms

3RT Three Rate Technology.

AGL Above Ground Level.

API Application Programming Interface.

APS Aerial Photo Survey.

AVIRIS Airborne Visible Infrared Imaging Spectrometer.

B boron.

BBA Block Bundle Adjustment.

CARI Chlorophyll Absorption Ratio Index.

CRP Calibrated Reflectance Panel.

CRT Constant Rate Technology.

Cu copper.

DLS Downwelling Light Sensor.

DSM Digital Surface Model.

DTM Digital Terrain Model.

ECC Enhanced Correlation Coefficient.

ENDVI Enhanced Normalized Difference Vegetation Index.

Fe iron.

FLANN Fast Library for Approximate Nearest Neighbors.

FOV Field of View.

GIS Geographic Information System.

GPS Global Positioning System.

GUI Graphical User Interface.

HFOV Horizontal Field of View.

I iodine.

IMU Inertial Measurement Unit.

IOT Internet of Things.

LAI Leaf Area Index.

LWIR Longwave Infrared.

MCARI Modified Chlorophyll Absorption Ratio Index.

Mn manganese.

Mo molybdenum.

MSAVI Modified Soil-Adjusted Vegetation Index.

MSE Mean Squared Error.

MSMS MultiSpectralMicroSensor.

MSR Modified Simple Ratio.

MU Management Unit.

NDVI Normalized Difference Vegetation Index.

NIR Near-infrared.

OpenCV Open Source Computer Vision Library.

PA Precision Agriculture.

PDR2020 Programa de Desenvolvimento Rural 2014-2020.

PWM/GPIO Pulse Width Modulation/General-Purpose Input/Output.

RC-Helicopter Remotely Controlled Helicopters.

RDVI Renormalized Difference Vegetation Index.

RE Red-Edge.

RGB Red, Green and Blue.

RICS Robotic & Industrial Complex Systems.

ROA Remotely Operated Aircraft.

ROS Robot Operating System.

RPV Remotely Piloted Vehicles.

RTK Real Time Kinematic.

SAVI Soil-Adjusted Vegetation Index.

Se selenium.

SfM Structure From Motion.

SR Simple Ratio.

SWIR Short Wave Infrared.

SWIR-2 Short Wave Infrared 2.

SWIR-1 Short Wave Infrared 1.

TVI Triangular Vegetation Index.

UAS Unmanned Aerial System(s).

UAV Unmanned Aerial Vehicle(s).

VRT Variable Rate Technology.

VTOL vertical take-off and landing.

Zn zinc.

Nomenclature

Some of the most common and constant symbols used throughout the thesis are listed below.

Units

g	Gram
kg	Kilogram
m	Metre
cm	Centimetre
mm	Millimetre
nm	nanometre
s	second
sr	steradian
W	watt
MB	Megabyte
Gb	Gigabit
px	Pixel
E	Irradiance
L	Radiance

CHAPTER 1

Introduction

1.1 Context and Motivation

In today's world, technology is increasingly becoming part of human life; in an environment where data acquisition is an issue in our community since information is continually growing and changing. People no longer need to learn how to adapt to innovation but instead technology is responsible for modelling itself around society; by processing all the sensor's acquired information, getting results, conclusions and even learning how to behave even when faced with unknown situations.

Technology enhancement is embracing society, not only by learning how to adjust itself to each individual and their own personal devices, but also by reaching various industrial sectors in great scale. Fields of operation where UAV can give a huge contribution, achieving as much information as possible in the least possible time about any Big Data systems, such as traffic surveillance, security and rescue operations, building inspections, mapping or even agricultural monitoring.

Nowadays, one of the biggest concerns in our society is related to both food and water security, as well as well-nourished population assurance, which involves selling food that provides the major nutrients to preserve a healthy society.

This thesis' concept urged from various projects related to fruit and cereal cropping fields, technologically arising in partnership with Faculdade de Ciências e Tecnologias (FCT-UNL), such as:

- MPBIO – GO Biofortificação de Tomate para Processamento Industrial e em Modo de Produção Biológica (Biofortification of Tomato with Mg, Zn e Fe) – Project Number PDR2020-101-030701, 01 Ação 1.1/2016, Initiative nº 6, Partnership nº 11;
- BATCAL – GO Biofortificação de Batata em Calcio (Biofortification of Potato with

Ca) – Project Number PDR2020-101-030719, 01 Ação 1.1/2016, Initiative nº 20, Partnership nº 17.

- PERCAL – GO Biofortificação de Pera Rocha em Calcio (Biofortification of Pear with Ca) – Project Number PDR2020-101-030734, 01 Ação 1.1/2016, Initiative nº 148, Partnership nº 76.
- UVAZN – GO Fortificação de uva em zinco para vinho branco e tinto (Biofortification of Grapes with Zn) – Project Number PDR2020-101-030727, 01 Ação 1.1/2016, Initiative nº 144, Partnership nº 74.
- INSTAGRI - Intelligent Cloud Based Environment for Precision Agriculture using Remote Sensing Technology – Project Number CENTRO-01-0247-FEDER-023152.

The previously mentioned projects rely on new Portuguese motivational researches of an agricultural change in the country. It has been gradually growing in Portugal for several years, through national companies like Syngenta or Wisecrop, already with creative and efficient developed solutions in the market. They use techniques based on aerial multispectral captured images of the field area, capable to gather information invisible to human eye, in order to advise farmers about their crops growth, indicating possible plagues, diseases or even fields' malnutrition.

1.2 Goal and Approach

Despite the distinct techniques taken by different entities on the agricultural business, this thesis proposes an open-source framework development for precision agriculture that will be fully integrated with an UAV, which is the main asset responsible for the agricultural field autonomous scanning.

With the use of an incorporated multispectral RedEdge-M camera, developed by Micasense, the UAV intends to collect a full real-time imagery dataset of the cultivation area, which will provide five different colour spectrum bands for each taken frame: Red, Green, Blue, Infrared and Red-Edge.

Furthermore, it should allow an index fusion between the bands and the information collected by the camera using the Open Source Computer Vision Library (OpenCV), a major tool for image and video analysis. The multispectral imagery effects over the overflown area ground type will provide feasible information about possible terrain irregularities, obstacles or target fertilization points, for instance. Thus, the imagery revealed features will be compiled into a layered map with certain areas labelled according to the terrain type detected. This will then be fused into an occupancy map, in order to support the UAV autonomous navigation in constrained environments.

This thesis analysis and monitoring system goal is supported with the ability to interpret the different effects caused on each captured field area's colour band, due to the type of terrain features.

This approach intends to improve the acquired knowledge about Precision Agriculture, as well as autonomous navigation and multi-robot cooperation. It is an opportunity to share and enhance the Robotic & Industrial Complex Systems (RICS) research group with the explored and developed material about the field. Therefore, this dissertation is focused on analysing and enriching the usage of Multispectral cameras in this industry.

1.3 Dissertation Structure

This dissertation presents a fully integrated system for terrain classification and mapping. It is organized into 6 chapters along with this section, starting from the state-of-the-art review until the actual project implementation and it's final statements and conclusions:

- **Chapter 1: Introduction** presents the project and main goals to achieve with terrain classification and mapping, as well as multiple projects that motivated this thesis' development;
- **Chapter 2: State of the Art** shows the UAV usage background around agricultural sector and the multispectral imagery evolution throughout history, reaching nowadays drones advantages in daily tasks;
- **Chapter 3: Supporting Concepts** enrols the reader with basic technology concepts integrated in this project;
- **Chapter 4: Proposed Framework** describes the framework and introduces the infrastructures applied in the scope of this project;
- **Chapter 5: Framework Implementation** details the implementation steps;
- **Chapter 6: Experimental Results** includes the model parametrization and testing, containing the output results, their validation and analysis;
- **Chapter 7: Conclusions and Future Work** sums up all the implemented strategy, challenges, future developments and suggestions to improve the algorithms and project ambitions;
- **Appendix A: Rededge-M Imagery Sensor** Sample of a YAML metadata file from Rededge-M Imagery Sensor;
- **Appendix B: Terrain Classification Results** Tables including testing overall results.

CHAPTER 2

State of the Art

2.1 Challenges in Modern Agriculture

Nowadays, with environmental change, it becomes challenging for society to embrace smarter ways for crop improvement or water and fertilizers control. Underdeveloped countries are increasingly facing agricultural salinity and waterlogging problems due to the lack of irrigation management. This same lack of water issues are starting to manifest crop production limitations in developed countries [1].

According to David Tilman, it was studied that in approximately 30 years the population will increase 35%, meaning that crops' production should double [2]. Besides, growth in demand for protein food is also a concern since the pace of production must overcome the demographic growth (Figure 2.1).

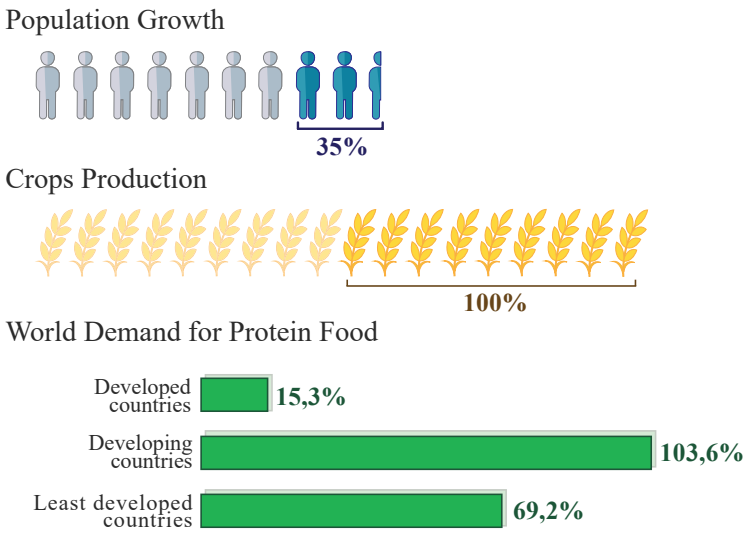


Figure 2.1: Agricultural needs due to population growth, in 2050. Adapted from [2].

According to [2] agricultural nourishing issues can be mitigated or even eradicated by taking the following 5 actions:

1. Stopping Agriculture's climate footprint → Preventing deforestation when building new farms;

2. Harvesting increase in the existing farmlands → Enhancing farms' production by adopting new technological systems capable to accurately adjust the use of fertilizers and pesticides;

3. Ensure a greater efficiency of the resources → Agricultural practices using resources capable to adjust according to the field characteristics and state of production;

4. Diet change → Since, only 55% of the crops' calories are ingested by humans, whereas the rest goes to cattle, biofuels and industrial products, it would be ideal to change some population food habits. However, this is considered an almost impossible measure to fulfill, which therefore leads to the adoption of nourishment crops for the production of bio-fuels as a solution to enhance food availability;

5. Reducing the food waste → Minimizing the loss of food before consumption not only in the developed countries where there is an avoidable excess of wasted food at supermarkets, restaurants and residences, but also at the least developed countries where the products loss due to transportation issues is still too high.

Regardless, food consumption still grows according to the associated population increase, as well as urbanization considers people's preference on moving from their rural livelihoods to bigger urbanized cities. Both issues are correlated since urbanization limits food production, which therefore leads to tropical forests' destruction aiming to increase the cultivable soil.

Agriculture is one of the main causes for global warming due to deforestation (which secures biodiversity and natural carbon cycle) and greenhouse gas emission that represents proportions even bigger than cars, trains or aircraft emissions [2]. It is critical to find a solution for crops' production increase within the limited land available and water usage restrictions for environmental causes. Besides, reducing energy consumption is a necessity as well, such as the continuous food production taking into account the inevitable climate change impacts already caused [1].

On the other hand, along with the food production it is also very important to keep the high nourishment levels; for instance, article [3] studies the nutritional deficiency at China's soil and population, derived from the lack of elements in food (Figure 2.2). As an example, the eminent lack of iron (Fe) evidences itself in the anaemic population increase, sub-clinical deficient levels of boron (B), copper (Cu), manganese (Mn), molybdenum (Mo), zinc (Zn), selenium (Se) or even iodine (I) affects millions of people with goitre disease.

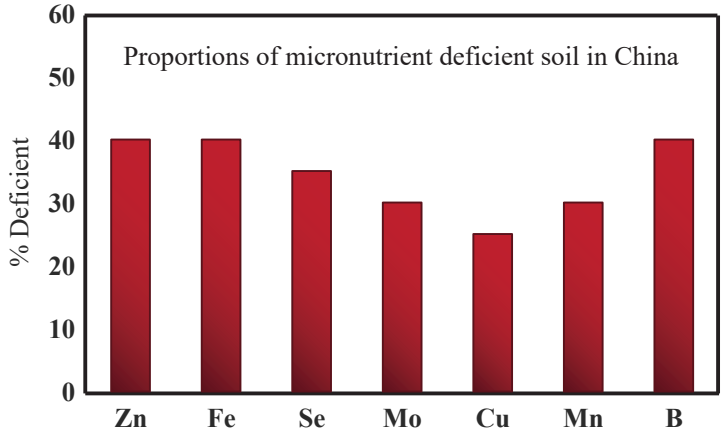


Figure 2.2: Proportions of micronutrient deficient soil in China. From [3].

2.1.1 Fruit's Biofortification in Agriculture

In China, according to [3], 40% of the land is clearly deficient on Zn and Fe, therefore it is critical to find plant nutritional strategies capable of solving these issues. In particular, the biofortification approach is considered sustainable, low-cost and reveals to be efficient increasing the breeding of crops' micronutrients density and bioavailibility. Since China's nutritional regime mainly includes tube crops and vegetables, the enrichment of plants, rice, wheat and maize is a priority.

Recently, there were also analysed the agricultural practices with micronutrients' fertilizers. On the other hand, article [3] reviews two fertilizer types: soil and foliar application. Despite the major nutritional effects in micronutrients's density and plant's bioavailability, it also refers to the drawbacks around the large temporal and spatial recovery due to the fixed plant-unavailable form issue behind soil application fertilizers, and to the inefficiency of foliar-applied micronutrients since the plant roots are not usually reached.

Therefore, farming tends to improve with breeding programs' implementation, field drainage and a better fertilizer technological strategies usage. Indeed, it is critical to reduce the over-fertilization and pesticides usage to lower the global greenhouse gas emission levels, the degrading of water resources or even farmland and aquatic ecosystems impacts [1].

2.1.2 Alternative Application Systems to Prevent Over-Fertilization Issues in Agriculture

Variable Rate Technology (VRT), Site-Specific Farming or Precision Agriculture are some of the nomenclatures given to the optimal fertilization supplement that instead of being a nutrients' single rate based process (Constant Rate Technology), it will be able to accordingly adjust itself to each location needs (Multiple Rate Technology) [4].

However, there are some apprehensions around the agricultural developed equipment, currently capable to regulate the under/over-fertilization issue in the agricultural field.

2.1.2.1 Implementation's Economic Feasibility/Viability (Business Case)

The efficiency impacts of this nutrients non-waste process are remarkable, however the article [4] reveals an interesting study of nitrogen fertilization on corn that answers to the real question around VRT investment: the balanced trade-off between the revenues and the costs of application involved. In order to proceed with this study, it also took into account the Constant Rate Technology (CRT) and the Three Rate Technology (3RT) study cases, which provided the ability to compare the net revenues between the three technologies (VRT, CRT and 3RT).

Tests were also made in two distinct types of fields: one area with an average fertility level homogeneous along the whole field, and another area less homogeneous. In other words, whereas field A was spatially homogeneous, field B had a greater soil spatial variability of fertility levels.

Simultaneously, it is a fact that the less fertilized is the field, the greater will be the efficiency gain with the CRT method. However, VRT provides a proper nitrogen quantity until a fertility Management Unit (MU) of 1.5m (minimum MU possible).

On the other hand, this area of production detailed analysis requires a certain investment in more complex equipment capable of handling smaller management unit area and greater information volume to be analysed. In return, there is a cost decrease referred to the less wasted fertilizer and yield increase.

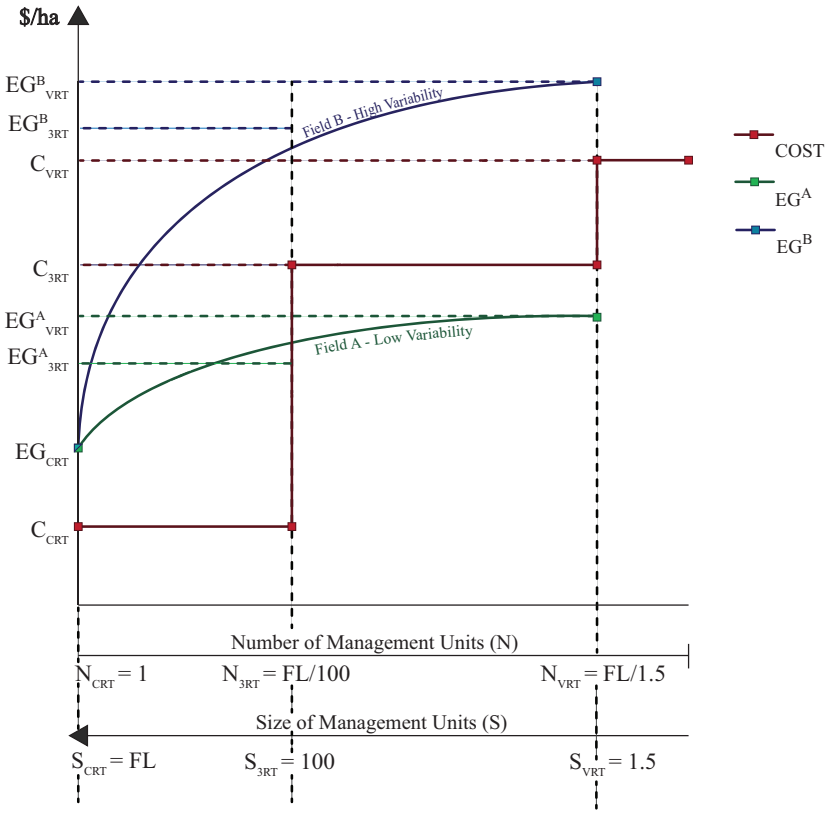


Figure 2.3: Efficiency gains (EG) and application costs (C). Graph adapted from [4].

Although the positive results revealed in the article due to the studied methods' efficiency gains versus associated costs (Figure 2.3), the CRT method was more profitable in small fields with an homogeneous fertility than VRT because when there is a field with an average fertility high but with a low variability, it means that more areas require the maximum possible rate of fertilizer which will therefore generate the best yield gain. Furthermore, the revenues of implementing a VRT system under the previous conditions wouldn't be sufficient to cover the application costs.

Nevertheless, the multiple rate technology (VRT) was more profitable in most of the scenarios in study, with larger net revenues than CRT or 3RT, and besides, environmental benefits of VRT were notable as the variability of the studied area's fertility increased comparing to the CRT environmental impacts due to the excess of fertilizer in use.

2.2 Remote Sensing in Precision Agriculture

Precision Agriculture (PA) is a technology that has been gradually growing over the years, beneficing farmers who need to regularly manage their agricultural fields. The application of several sensors, strategically placed and distributed over the field was a step-forward into the autonomous analysis of terrains, since it became possible to detect the field changes and retrieve feasible information about its state of production nearly in real-time.

Correlation between multi-spectral imagery and geospatial information from a Global Positioning System (GPS) or a Geographic Information System (GIS), was pivotal to foster the evolution of this technology. It allowed mapping between the image data and the geographical location of its acquisition (i.e. image georeferencing). However, only with the appearance of remote sensors with enough quality and reasonable costs, arose the opportunity to finally invest in the agricultural sector, since until then the only collected imagery data used for this purpose mostly came from satellites, which had high-resolution, but were an expensive approach, as well as limited accessibility.

Remote sensing applications in PA are also able to regularly provide farmers feasible information about their field, which can possibly reveal critical crop conditions still non-detectable by human eye but technologically collected in advance through complex equipment. The multispectral imagery output will indicate different index values at the distinct bands, from which data analysis can reveal different meanings about the crops growing state.

Therefore, remote sensing grant a better agricultural monitoring and the possibility to take actions in a timely manner in order to accordingly adjust the undertaken procedures and prevent associated losses [5].

The article [5] also refers the four different stages of PA process:

1. Data collection $\rightarrow$ sensing and monitoring information about Leaf Area Index (LAI), surface soil properties, water stress, nitrogen content, crop height, ground biomass, crop yield, weed extent, crop species, leaf chlorophyll content;

2. Mapping → field variation, crop-type, tillage, crop residue and yield mapping;

3. Analysis and decision making → calculate the finest imagery acquisition frequency, control the fertilizer quantities and other decision making procedures;

4. Performance management → intervention over the plans and established implementations to correct the farming methods under process [6].

According to [5], remote sensing could autonomously be responsible for the previously mentioned first three PA steps, efficiently collecting large amounts of data from extent agricultural fields within a short period of time.

Along with the expansion of multispectral technology and data fusion, there are various monitoring applications arising that, not only take near real-time farm analysis as a priority, but also motivate the environmental care within the agricultural sector.

2.2.1 The Application of Unmanned Aerial Systems in Precision Agriculture

At the moment, VRT combined with GPS is one of the most advanced technologies in precision agriculture, but for its entirely success, as previously stated, complex equipment and high quality sensors are required in order to capture detailed maps of the entire field in study and capable to accurately detect possible plagues, diseases or malnutrition.

Unmanned aerial systems are the main competitors in the market against satellites' imagery acquisition due to its accessibility efficiency (e.g. daily or even near real-time monitoring), detailed resolution (e.g. centimetres) and reasonable costs involved. Unmanned Aerial System(s) (UAS) empower privileged views over the crops and passiveness against weather conditions (e.g. clouds or fog), unlike satellites or manned aircraft.

At the beginning, the exploring and investigation around UAV - vehicles without on-board control - for photometric purposes, started by implementing kites, blimps and balloons, but despite the lower costs involved, they were too big and therefore considered difficult platforms to maneuver. Accordingly, smaller vehicles became desirable, for instance Remotely Piloted Vehicles (RPV) and Remotely Operated Aircraft (ROA).

a b

Figure 2.4: Radio-controlled UAV used in [7] to test the agility and flight constraints of the vehicles in study: (a) L'Avion Jaune's powered glider, (b) Pixie motorized parachute.

In article [7], two different kinds of vehicles are mentioned for testing their manoeuvrability (figure 2.4), flight restrictions and abilities to accurately capture multispectral imagery data from the crops. They were both incorporated with digital photographic cameras capable to extract the visible and near-infrared bands from the spectral filters in use.

Nevertheless, the previous study revealed that UAV require lower altitudes during flights to achieve higher spatial resolutions, which means that there may be more instability and vulnerability when acquiring different resolutions and distinct viewing angles between the shooting images. This can possibly be corrected by warping the collected pictures based on each georeferenced information.

In article [8] a ground classification of a certain field in Mexico was produced using a fixed-wing aircraft to capture the imagery, using an incorporated frontal video camera at the nose of the UAV and a digital camera at the aircraft's left wing (Figure 2.5 (a)).

On the other hand, article [9] enhances this thesis with an historical research about RC-Helicopters. [9] not only points to the photogrammetry problems due to the UAV vibrations caused by the rotors, but also considers the GPS accuracy as a solution.

a b

Figure 2.5: (a) Fixed-wing aircraft (BAT 3 UAS), from [8]. (b) Remotely Controlled Helicopters (RC-Helicopter) by Helicam and weControl, Switzerland, from [9].

Even so, unmanned helicopters are valuable covering small terrains and the costs involved are even lower than aircraft investment.

Recently studies around UAV in PA were taken in the agronomic R&D plant of Syngenta Crop Protection AG in Stein (Canton AG, Switzerland). The MultiSpectralMicroSensor (MSMS) project was divided into two phases where the main goal was to research about light-weight multispectral sensor's prototype incorporated with two distinct UAV: firstly with an autonomous mini-helicopter developed by weControl and then with a micro-UAV.

Both studies intended to capture the Red and Near-infrared (NIR) multispectral domains of the grapevine in research and, after that, asses the field's imagery processing, from which was possible to match the Normalized Difference Vegetation Index (NDVI) calculated values with the damaged leaves' percentage.

a

b

Figure 2.6: (a) Helicopter based Mini-UAV by weControl (Zurich), from [10]. (b) Quadcopter micro-UAV 'microdrones md4-200', from [10].

The quadcopter in Figure 2.6 (b) was favoured in respect to the one represented in 2.6 (a), not only due to its lower weight and cost attributes, but also to its vertical take-off and landing (VTOL) capability, giving a huge stability effect to the flight, which in turn improves the quality of the captured images.

2.2.2 Multispectral Sensing

When it comes to multispectral sensing subject, it is important to be aware of the different concepts around illumination in order to understand a few but very important details about the terms in use, described in [11]:

- Irradiance → Refers to the light energy emitted into the surface area, per time unit. Measured in watts per square meter (W/m^2)
- Reflectance → It refers to values ranged between 0 and 1 to characterize the incident light reflected by a surface. This fractionated value can be parametrized into distinct variables such as the reflected light wavelength or even the incidence and reflection angles.
- Radiance → It is a normalized value of the irradiance intensity according to a certain solid angle, typically specified in steradians (sr) and accordingly variable with the light's propagating direction. Typically measured in $W/m^2/sr$, but when related to the spectral domain it may be measured in line with the wavelength $W/m^2/sr/\mu m$.

According to [11], there are several ways to contactlessly acquire information about a certain object or scene, relying on the sunlit reflected light detected by electro-optical sensors capable of measuring the different wavelengths within the visible spectrum: Panchromatic band (i.e. grayscale) or colour imaging sensors (i.e. Red, Green and Blue (RGB)). Besides, there are also thermal sensors, that instead of being light-based, they acquire information relying on the body temperature emissions (i.e. Longwave Infrared (LWIR)).

Nevertheless, there has been an imaging evolution starting to spread its horizons into new bands apart from the RGB and the grayscale domains, by adding for instance, NIR and Short Wave Infrared (SWIR) bands which, when fully integrated and studied together

may result in valuable information about characteristic object properties according to its molecular composition, differently exposed through the electromagnetic spectrum.

Therefore, it is possible to achieve some conclusions from multispectral cameras' acquired images such as revealing the type of material from which it is composed or, at least, distribute the objects in scene by groups, relying on their material classification properties.

However, in order to take full advantage of the available bands, multispectral imaging gets through a spectral processing, also know as spectroscopy that encompasses the measurement, analysis and finally the result's interpretation, which can possibly reveal valuable information about the scene in analysis, [11]. On the other hand, when it comes to large scale areas, it is called imaging spectroscopy, as illustrated bellow in Figure 2.7.

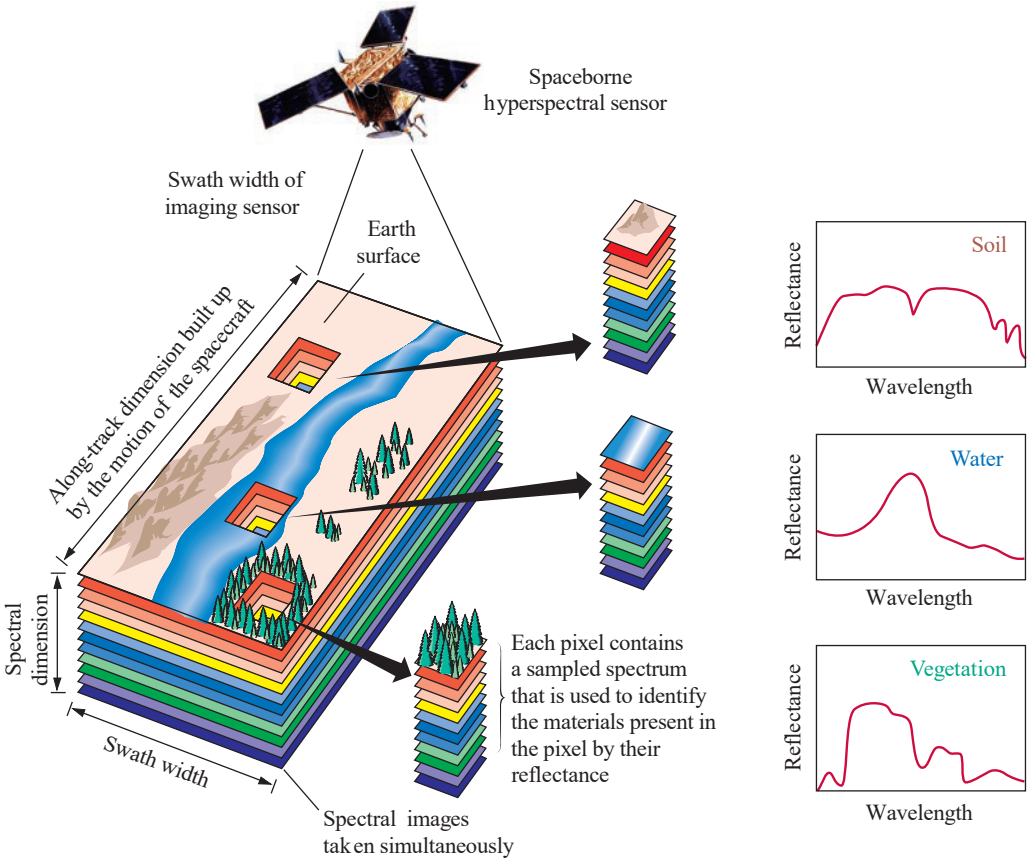


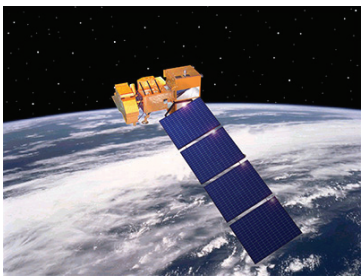
Figure 2.7: The concept of imaging spectroscopy. Illustration from [11].

Imaging spectroscopy retrieves some samples from the ground-based scene where each pixel from the processed image corresponds to the reflectance measured value of the exploited area. For example, in Figure 2.7 the graphs on the right represent different spectral variances between the surfaces encountered in the scene, in this case: soil, water and vegetation; from which it was possible to classify the type of terrain, [11].

Earlier, as illustrated in figure 2.7, multispectral imagery solely derived from satellites. Article [12], for example, describes a study made in Mozambique from where the multispectral imagery dataset in analysis was collected by two different satellites in order to evaluate

the terrain conditions along the years. Figure 2.8 (a) represents Landsat 5 satellite from which 1989 imagery was collected, and Figure 2.8 (b) shows Landsat 7 satellite that later obtained the dataset from 2002 and 2005.

a



b

Figure 2.8: Satellites used as multispectral imagery acquisition sensors in [12]. (a) Landsat 5, from [13] and (b) Landsat 7, from [14].

In this research example there were only used five of the seven available bands of the spectral domain: Green-Band, Red-Band, Near-Infrared, Short Wave Infrared 1 (SWIR-1) and Short Wave Infrared 2 (SWIR-2). However, although never used in [12], Landsat 7 could already provide information within the thermal band (band 6 – Spatial Resolution: 60 m/pixel), the panchromatic band of the electromagnetic spectrum (band 8 – Spatial Resolution: 15 m/pixel) and the spectral Blue-Band (band 1), with a spatial resolution of 28.5 m/pixel, similarly to the ones in use.

Furthermore, article [15], also enhances the spaceborne sensor-based studies with new spectral imagery researches about vegetation classification, which beyond the Landsat TM, also studied other satellites' datasets such as SPOT-XS, NOAA-AVHRR and IKONOS. From this research there were obtained reasonable results about mapping vegetation species based on the collected Vegetation Indices.

However, in [15], there were described certain sensors' constraints requiring improvements not only related to the limited spatial and spectral sensors' resolution, but also to the arising questions about the uniqueness of the reflectance spectral properties of a plant, since apart from the specie, there are other variables due to their age, atmospheric and soil growth conditions, topography or plant stresses, for instance.

On the other hand, in 1987, spectral imaging stepped into the hyperspectral domain with the appearance of the first airborne imaging sensor released by NASA Jet Propulsion Laboratory (JPL), the Airborne Visible Infrared Imaging Spectrometer (AVIRIS). Whereas the multispectral domain was limited between 3 to 10 wider bands, the expansion into the hyperspectral domain, with hundreds of narrow available bands, significantly increased the imagery sensors' accuracy [16].

The spectral complexity around the AVIRIS's collected imagery opened up an opportunity to distinguish different minerals solely based on the Reflectance Spectrum, since each mineral has a different molecular constitution possible to detect by the distinct regions where the mineral absorbs the energy, [17]. For instance, as described in article [15], it was possible to distinguish some salt marsh species: Salicornia, Grindelia and Spartina.

Despite hyperspectral systems' accuracy, they are still an expensive approach and timely-consumer due to the high-resolution processing involved [11].

For that matter, according to [9], in 1980 the first helicopter for photogrammetric purposes appeared, capable of reaching a maximum payload of 3 kg, which means that by attaching cameras such as Rolleiflex SLX or Hasselblad MK 20 it became possible to acquire some aerial imagery from the helicopter.

On the other hand, with the appearance of smaller aerial systems, such as the UAV, and their continuous evolution into becoming increasingly lighter, it also brought the drawbacks related to the vehicle's possible payload, limited to approximately 20-30% of the system's total weight [10]. The physical attributes of the multispectral imagery sensors were also a main factor to the continuous study around this science, which led to a natural adaptation of these sensors and therefore cameras' technological research led to their weight decrease.

The imagery sensors used in [8], a Canon EOS-350D and a Sony DSC-F828, were coupled to the vehicles illustrated in figure 2.4 in order to be tested and compared with some of the satellites previously mentioned within the multispectral domain. Despite being simple digital cameras only collecting information from the visible spectrum, whereas Canon camera's CCD-matrix could split the light into three channels the RGB, Sony was divided into four, adding the Cyan to the provided bands.

Even so, they had to be slightly modified by changing the band-pass filter in front of the CCD-matrix, sensible to infrared radiation up to 900 nm, but changed by a high-pass filter attached to the camera's objective in order to narrow a new band between the 720 and 850 nm, which approximately represents the NIR band of the common satellite sensors, only shifted by a mere 50 nm. This research improvement allowed to encompass the main infrared bands reflected by vegetation and therefore proceed with studies around precision agriculture [8].

In [10], lighter cameras were used, the mini-UAV helicopter was attached to a Sony Smart Cam (NIR sensor) and a Canon EOS 20D (RGB sensor), but when testing the micro-UAV, although limited to the Red a NIR bands, it was incorporated with the MSMS sensor (total weight of 350 g).

Furthermore, both in [18] and [19] imagery data from a 6-band multispectral camera (MCA-6, Tetracam, Inc., California, U.S.), with an image resolution of 1280×960 pixels, is taken into consideration. Those sensors, with the band-set properly configured (according to Table 2.1), are able to graphically illustrate vegetation indices.

Filter Designation	Band Name	Filter Applications
490FS10-25	Blue	1. High penetration of water bodies - capable to distinguish soil and rock surfaces from vegetation. Useful to detect cultural properties; 2. This band is sensitive to loss of chlorophyll, browning, ripening, senescing, and soil background effects; 3. Excellent predictor of grain yield.
550FS10-25	Green	1. This band is sensitive to water turbidity differences; 2. Useful at predicting chlorophyll content.
680FS10-25	Red	1. One of the main variables to calculate NDVI, from which is possible to detect changes in biomass, LAI soil background, types of cultivation, canopy structure, nitrogen, moisture, and stress in plants.
720FS10-25	Red Edge	1. Describes the vegetation spectral reflectance curve between 690 nm and 740 nm - caused by the transition from chlorophyll absorption of red wavelengths and near-infrared reflection due to the mesophyll cells in leaves which in healthy plants act like a mirror to NIR. 2. This band is sensitive to temporal variations in crop growth and vegetation stress. 3. Provides additional information about chlorophyll and nitrogen status of plants.
800FS20-25	NIR	1. One of the main variables to calculate NDVI, together with the Red band. It is also used for determining the RDVI (related to the maximum absorption in the red caused by chlorophyll pigments and the maximum reflection in the infrared due to leaf cellular structure), MSAVI (developed to cancel soil reflectance), SARVI (minimizes both canopy background and atmospheric effects). 2. Variable to calculte various plant pigment ratios such as: Pigment Specific Simple ratio Chl (PSSR), Pigment Specific Normalized Difference (PSND) and Structure-Insensitive Pigment Index (SIPI) - These ratios can be related to vegetation stress conditions.
900FS20-25	NIR	1. This is the maximum peak of the NIR spectrum. It can reveal critical information about the crop's type/growth stage. 2. Useful for computing crop moisture sensitive index.

Table 2.1: MCA-6, Tetracam, Inc., California, USA multispectral sensor band-set applications. Adapted from [20].

Moreover, the canopy reflectance within each band, from both visible and infrared region, are highly considered to describe the vitality of a plant, as illustrated in figure 2.9.

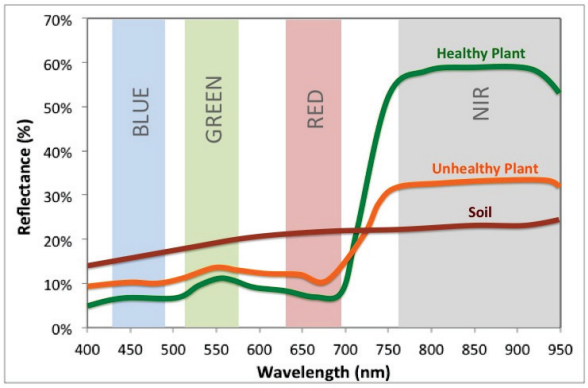


Figure 2.9: Reflectance spectrum of a typical green leaf. Illustration from [21].

In essence, as described in Table 2.1 and illustrated in Figure 2.9, there is much information to be extracted from the multispectral range of available bands, at each band's individual level and, above all, the acquired data fusion and its further processing can be highly profitable to the agricultural sector, by giving feasible updates about the crops' state of growth, plant stress or vegetation quality.

2.2.2.1 Index Fusion

Article [22] is an overview about a project centralized in different existing vegetation indices studied, such as NDVI, RDVI, SAVI, MSR, MSAVI, TVI, MCARI; a diverse set of reflectance graphical leaf area index (LAI) indicators.

Green LAI is a very helpful variable for precision agriculture since it returns the green leaf area according to the ground surface area. From this information it is possible to make an evaluation about the foliage cover in a certain surface from which physiologists are able to acquire some results and make a crop growth and yield prediction, providing a considerable overview about the field's productivity.

Furthermore, it is also interesting to understand the biophysical exchanges between the vegetation and the atmosphere (CO2 exchange, plant transpiration), as well as the LAI obtained only based on the plants' pigment contents (e.g. chlorophyll).

This study encouraged many researchers into the discovery of new algorithms and different techniques not only to return the canopy multispectral reflectance index from large agriculture fields, but also to deepen the spectral fusion analysis. It led to a correlation discovery between LAI and other atmospheric factors, such as the shadows or soil brightness due to sunlit factors, as well as the spectral reflectance indices' investigation to acquire a lot of information from the fields productivity and crops' health state. It is important to consider that there is still no index capable of collecting only the information about one specific variable of interest or filtering the others reflectance influence.

Rather than retrieving each variable information such as the vegetation pigment content, getting the structural geometry of a plant, or canopy architecture; the indices were used to capture some natural leaf process results and optimize them to assess a certain purpose.

However, beyond the fact that canopy reflectance manifests a clear influence related to the structural (e.g. LAI), and biochemical properties such as chlorophyll, in the visible and near-infrared (NIR $\rightarrow$ 800 nm) bands, it also presents similar effects on the green and red reflectance spectral region (670 nm). These fusion effects given rise to techniques capable of filtering the chlorophyll properties from the LAI content influence, on the other hand, there were no studies referring to decoupling the LAI reflectance response from the chlorophyll's effect. The estimation of LAI without chlorophyll interference presents issues not only due to the saturation level reached between 2 and 5 but also because there isn't a unique relationship found between LAI and vegetation index of choice. For this reason, the various vegetation indices were analysed in order to detect which one provided the best results accordingly to LAI and which ones were less affected by the external effects such as atmospheric conditions, spectral reflectance of the canopy, illumination geometry or namely soil optical properties.

The Normalized Difference Vegetation Index (NDVI), equation 2.1, is the most well-known and used reflectance index: "It is based on the contrast between the maximum absorption in the red due to chlorophyll pigments and the maximum reflection in the infrared caused by leaf cellular structure." [22].

$$NDVI = \frac{NIR - RED}{NIR + RED} \tag{2.1}$$

Equation 2.1: NDVI equation.

Although NDVI is one of the most used indices, it is mentioned that it saturates in dense multilayered canopy and also that it presents non-linear shapes in relation to biophysical parameters (eg.: green LAI). For that reason, there were improved new algorithms such as the Renormalized Difference Vegetation Index (RDVI), the Enhanced Normalized Difference Vegetation Index (ENDVI) and the Modified Simple Ratio (MSR) in order to linearise it due to the vegetation biophysical variables.

$$ENDVI = \frac{(NIR + GREEN) - 2 * BLUE}{(NIR + GREEN) + 2 * BLUE} \tag{2.2}$$

Equation 2.2: ENDVI is an improvement of NDVI where red and green are used as the reflective channels and blue as the absorption channel.

$$RDVI = \frac{NIR - RED}{\sqrt{NIR + RED}} \tag{2.3}$$

Equation 2.3: RDVI is the combination between DVI (NIR-RED) and NDVI, for high and low LAI values [22].

$$MSR = \frac{\frac{NIR}{RED} - 1}{\sqrt{\frac{NIR}{RED} + 1}} \tag{2.4}$$

Equation 2.4: Combination with Simple Ratio (SR): NIR/RED; Presents more Linear effects due to Vegetation Parameters [22].

There were also Soil-Adjusted Vegetation Index (SAVI) and Modified Soil-Adjusted Vegetation Index (MSAVI) algorithms created to determine soil optical properties with the minimum background effects.

$$SAVI = (1 + L) \frac{NIR - RED}{NIR + RED + L} \tag{2.5}$$

Equation 2.5: L: Vegetation density Factor; L=0.5 (Suggested by Huete) [22].

MSAVI algorithm returned an alternative to the L factor definition, by generating its own adjustment to vegetation index; Besides that, MSAVI presented the best LAI results since it was the less affected by the canopy parameters variation as well as the soil spectral properties, even in dense canopies.

$$MSAVI = \frac{1}{2}\left[2NIR + 1 - \sqrt{(2NIR + 1)^2 - 8(NIR - RED)}\right] \tag{2.6}$$

Furthermore, it was also obtained a red band correction (R_{rb}) effect by the blue channel incorporation in order to minimize the atmospheric induced variation in NDVI, considering $\gamma = 1$ [22]:

$$R_{rb} = R_r - \gamma\,(R_b - R_r) \tag{2.7}$$

The latter led into a SAVI variation, independent from the canopy background and the atmospheric effects:

$$SARVI = \frac{(1+L)\,(NIR - R_{rb})}{NIR + R_{rb} + L} \tag{2.8}$$

Equation 2.8: Soil and Atmospherically Resistant Vegetation Index [22].

Regardless the already existent LAI formulas, the research in [22] also led into new vegetation indices, by joining the green (550 nm) and the red-edge (RE, the near-infrared rising slope approximately between 700 and 750 nm) spectral regions. Beyond the discovered Chlorophyll Absorption Ratio Index (CARI) function, developed to calculate the absorption depth of chlorophyll in the soil, Modified Chlorophyll Absorption Ratio Index (MCARI) variation was taken into account in order to simplify the calculus involved, by setting the LAI, chlorophyll and chlorophyll–LAI into fixed values.

$$MCARI = \Big[(RE - RED) - 0.2\,(RE - GREEN)\Big]\frac{RE}{RED} \tag{2.9}$$

Equation 2.9: Considering LAI(67%), chlorophyll(27%) and chlorophyll–LAI(13%)

Similarly, there is an equation to measure the leaf pigments' absorbed energy based on a band's triangle between green, red and infrared regions:

$$TVI = 0.5\Big[120\,(RE - GREEN) - 200\,(RED - GREEN)\Big] \tag{2.10}$$

Equation 2.10: Triangular Vegetation Index (TVI)

TVI not only increases according to the rise of chlorophyll absorption, detected by the Red reflectance decrease, or even with high quantities of leaf tissue, represented by a near-infrared reflectance decrease. Furthermore, it is important to notice that a loss of reflectance within the green region may be caused by chlorophyll concentration increase.

Nevertheless, TVI and MCARI can be truly useful within vegetation analysis not only for being less influenced according to chlorophyll but also for its sensitivity increase of the reflectance within the green band. Besides, less influence from atmospheric or soil conditions is still an asset.

Finally, article [22] also suggests two newer versions for MCARI and another two for TVI. MCARI variations are new alternatives capable of reducing the effects caused by chlorophyll and instead increase the sensitivity to LAI changes, such as:

$$MCARI1 = 1.2\Big[2.5\,(NIR - RED) - 1.3\,(NIR - GREEN)\Big] \tag{2.11}$$

On the other hand, TVI alternatives, in order to improve LAI influence while filtering changes at a pigment level were slightly modified within the infrared variables of influence:

$$MTVI1 = 1.2\left[1.2\,(NIR - GREEN) - 2.5\,(RED - GREEN)\right] \tag{2.12}$$

Moreover, there were also developed another two formulas' variations from 2.11 and 2.12, through a new factor capable of adjusting soil influencing effects:

$$MCARI2 = \frac{1.5\left[2.5\,(NIR - RED) - 1.3\,(NIR - GREEN)\right]}{\sqrt{(2\,NIR + 1)^2 - (6\,NIR - 5\sqrt{RED}) - 0.5}} \tag{2.13}$$

$$MTVI2 = \frac{1.5\left[1.2\,(NIR - GREEN) - 2.5\,(RED - GREEN)\right]}{\sqrt{(2\,NIR + 1)^2 - (6\,NIR - 5\sqrt{RED}) - 0.5}} \tag{2.14}$$

According to the results article [22], 2.13 and 2.14 led into an overestimation of LAI due to this significant NIR reflectance influence.

In conclusion, studies along index fusion science remains in continuous improvement and under development since multispectral domain is still a source of research for vegetation analysis, bringing many benefits into precision agriculture. Nevertheless, it is highly required a background within imaging processing for multispectral remote sensing, since merging the information altogether from imagery sensors and making data fusion visually presented is also a great need.

2.3 Image Processing and Aerial Mapping in PA

Along with the advancement of technology and, particularly, the insertion of UAS in order to support the continuous growth of precision agriculture, it was crucial to gather a set of procedures capable of processing and analysing all the collected information by imagery remote sensing.

Therefore, it becomes necessary to acquire some practices to image processing, including methods to calibrate all the systems in use, further line-shift and vignetting correction, the assurance that all the collected bands meet the same geolocation among identical features (band-to-band registration), atmospheric adjustments and frame mosaicking [5].

Despite the existing manual procedures, the development of a sufficiently robust and autonomous system capable of managing and analysing all the information was highly considered. Besides, efficiently sending the results to farm growers and field specialists was also an improvement for possible lack of nutrients' predictions, plagues and diseases detection or crops' state of production analysis, near in real-time.

For this purpose, after imagery correcting methods within the area covered by the UAV, it is relevant to make a correlation between the multispectral frames taken, by joining

them, using a software capable of merging the imagery collection into a wide plan of the area in study, based on georeferenced positions.

Furthermore, along with the orthomosaics created for large sections of fields, it is possible to produce layered maps, conducting farmers into a specified aerial view of their crops' production by tags and labels selected to cluster the data returned from the UAV's flight over the field.

In line with [6], UAV' autonomous navigation could also be supported by enhancing the georeferenced orthomosaic maps with labelled areas according to weather conditions, soil irregularities and vegetation density, by fusing these probabilistic data into a semantic cost map.

Accordingly, the following sections describe some of the existing imagery processing techniques for aerial mapping.

2.3.1 Image Stitching and Mosaicking

The insertion of UAV in precision agriculture due to its economical feasibility, accuracy and high resolution at imagery acquisition was certainly a step-forward within agriculture development.

Despite small UAV' low altitude flights, up to approximately 100 m, providing limited fields of view, it is possible to acquire thousands of images from the over flown area and therefore build a unified mosaic from the field, by stitching the similar features from each frame, in order to make a wider view of the covered area [23].

For this purpose, Pix4Dmapper, Photoscan, Aerial Photo Survey (APS) are examples of some of the most well-known software for imagery processing collected by UAV. In [24] it is studied the mosaicking performance for each of these software.

Pix4Dmapper, developed by Pix4D, uses techniques based on aerial triangulation and Block Bundle Adjustment (BBA) in order to autonomously calculate the positions and orientation of each image. Then, accordingly to the obtained 3D point cloud, it is possible to connect those points and build a Digital Surface Model (DSM) of the target. The created mosaic is built by combining the imagery taken with DSM [24], (Recommends: 75% frontal and 60% side overlap [25]). Although Pix4D reference in [25] is applied to version 2013a, currently, the last Pix4Dmapper version is enhanced with Digital Terrain Model (DTM), volume calculation, topographic contour lines, thermography maps detailed to each pixel's temperature values and 3D texture model functionalities [26].

On the other hand, there is also APS, a software built by Menci Software, where mosaicking is autonomously originated from DTM, DSM, point cloud, orthomosaics and also topographic contour lines abilities. In addition, it is enhanced with a manual edition panel for later required corrections (Recommends: 80% frontal and 70% side overlap [24]).

Last but not least, it was also studied Photoscan performance, which is a software, developed by Agisoft, based on photo triangulation also capable of generating the covered

area point cloud, DTM, DSM and orthomosaic from multispectral imagery processing (Recommends: 80% frontal and 60% side overlap [24]).

a b c

Figure 2.10: Software orthomosaicking results by: (a) Photoscan, (b) APS and (c) Pix4Dmapper. Illustrations from [24].

Although the positive results from the three mosaicking software, Photoscan and Pix4Dmapper presented good rectification of the borders into the correct positions with a lower Mean Squared Error (MSE) than APS system. However, the APS tool providing a later processing ability for imagery correction and border alignments is highly considered, once the post-processing may present a final result visually better. Another conclusion is that a minimum of four overlapped images are necessary to acquire an accurate orthomosaic of the field, therefore, the overflown area should encompass an extra margin around the perimeter in study.

In line with [23], block bundle adjustment (BBA) is the most conventional mosaicking technique in the commercial software developed. It is also known as Structure From Motion (SfM), a computer vision technique based on mathematical methods capable of minimizing the misalignment errors regarding the analysed topography of the terrain.

BBA method is mainly composed by four steps:

1. Initial camera model based on GPS and Inertial Measurement Unit (IMU) $\rightarrow$ 3D scene points' transformation into the corresponding bidimensional image point, reasoned by collinearity equations [23];

2. Pairwise image registration → Correspondence between the similarity points of the overlapped images (tie points) and followed by the point coordinates' estimation (in common) within the tridimensional scene (x_i, y_i, z_i);

3. Scene Point Projection → Reprojection of the estimated scene point into the previously considered overlapped images. Determination of the uncertainty error due to the misalignment between the previously calculated tie points and the new projected coordinates;

4. Minimization of Linearised Projection Error → Mathematical algorithms to minimize the non-linear and linear problems. LM algorithm is the most popular to minimize the linear issues.

This process will be continuously iterated, by repeating the last two steps (3. and 4.), until the error $(E_{f,g,i})$ no longer decreases. However, accordingly to [23], although the BBA mosaicking technique is the most commonly used, it is concluded that despite being a process achievable for areas covered by few images, it is clearly not feasible for wider fields, where thousands of images need to be processed. The latter could mean a search space of N=21000 dimensions for 1000 images, equation 2.15.

$$N = 6p + 3q \tag{2.15}$$

Equation 2.15: Number of search space dimensions, where p is the number of images and q is the number of scene points (q=5000 is the minimum to obtain sufficiently accurate results with 5 scene points in each image) [23].

Therefore, the processing memory used is still too high and computational time consumed is not viable (varying according to $O(n^3)$, 2.11) [23].

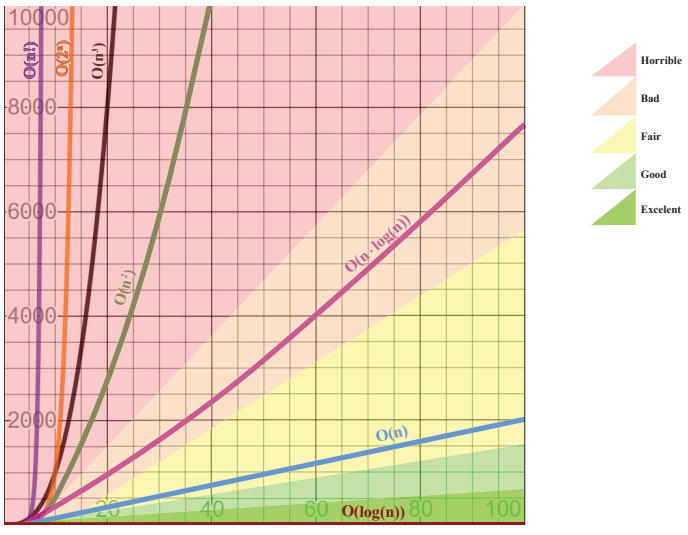


Figure 2.11: Big O notation. Adapted from [27].

This is why a faster and robust system was developed in [23], a BBA based algorithm improvement, where the first two steps of the new algorithm remain the same. On the other hand, the last two diverge from the previously mentioned in BBA method. The mosaicking process for "Quick Adjust" Algorithm consists in:

1. Initial camera model based on GPS and IMU;
2. Pairwise image registration;
3. Tie Point Back Projection (Intersection) → Triangulation method between the tie points determined within the overlapped images (i.e. intersection) by making a perpendicular back projection into the scene in study and estimating the coordinates of the previously calculated scene point (in step 1). The mean point between the determined coordinates represents the *back projection estimate* scene point (x_i, y_i, z_i)).
4. Camera Model Estimation (Resection) → Algorithms to find the camera model that minimizes the total back projection error through means of nonlinear least squares.

Similarly, this algorithm is also continuously iterated between the last two steps according to the new determined camera models until the total back projection error stops decreasing. It is then possible to create an accurate mosaic from surface model projected from the calculated scene points and their correlation with camera models.

Even though the third step of the algorithm is a very optimistic description of reality, since some of the tie points back projection may not converge in the nearby. In [27], this problem is solved with an estimate for the terrain elevation based on the weighted average between the z coordinate of the calculated back projection estimate and the scene point estimated in the previous iteration. This surface elevation estimation will also lead into a faster convergence of the algorithm.

In contrast to BBA algorithm, the "Quick Adjust" developed method is, not only capable of processing more images for wider fields, but also providing much faster results (varying according to $O(n)$ time, 2.11).

$$N = 6p \tag{2.16}$$

Equation 2.16: Number of search space dimensions, where p is the number of images [23].

Nevertheless, N, in equation 2.16, is a 6-dimensional space of p independent searches, meaning that an 1000 imagery dataset in "Quick Adjust" algorithm is much faster than in BBA, by reaching a time rate of 14 s per image [23].

In conclusion, imagery mosaicking enables building a field's reconstructed view. Further processing within the covered area mosaic, such as clustering the data into layered maps, to visually present unified information about the forecast field growth state of production is highly considered and a huge benefit for Precision Agriculture's evolution and an improvement within the management of wider cropping fields.

CHAPTER 3

Supporting Concepts

3.1 Frameworks and Computer Vision

For this purpose, it will be developed an open-source framework in Robot Operating System (ROS), precisely created for robotics applications, where the main code will be developed in C++, but also supported by some Python scripts as a bridge between the Micasense sensor and the framework in development for the imagery acquisition, processing and analysis.

The processing of the images in study will also be supported by OpenCV library which stores a bunch of open-source methods already developed by the community.

3.1.1 ROS

ROS is an open-source robot operating system. It is a framework that raised from other software prototypes created for particular robotic case studies or experiences with specific design goals. Ideally it should be a peer-to-peer, tools-based, multi-lingual, thin and free open-source framework. It was gradually built from all the challenges faced when developing several different projects of service robots. This framework concept, is not yet a reality, however it is important to be aware of its influence criteria through ROS' implementation, [28].

A peer-to-peer topology is required specially on robots where many messages are routed through a single wireless link which results in overflow requests colliding with each other. This issue requires a synchronization mechanism known in ROS as the named services or master module, responsible for the management of the bandwidth utilization, robots' stability and system's complexity.

Due to the whole existing robotics' software projects it is reasonable the re-utilization of drivers and shared algorithms. However, it is a fact that sometimes much of this code is

so entangled with the middleware that even extracting a single function from it can become a real "nightmare", not mentioning that many programmers have a certain programming language preference, which results in the great need to create a set of libraries capable to support a cross-language development interface where inputs are serialized through small executables and deserialized through those complex "black boxes" already prepared to analyse and process this kind of data with a neutral language, holding the ability to interpret mixed and matched languages like C++, Python, Octave and LISP.

ROS requires a certain knowledge of concepts based on communication nodes (software modules). There are two well-known interaction mechanisms based on messages (Asynchronous communication) or services (Synchronous communication). Regarding to the first one, nodes are known as publishers when they advertise some structured message to a certain topic (the link between two nodes communication). The node who listens to that certain kind of data is known as the subscriber. However, if it is required a request/response architecture, the service-based mechanism is the most desirable one because of its synchronous response dependent actions.

Furthermore, ROS framework presents an efficient debugging method, a great experimental procedure based on logged sensor data or played back launch processes through different graphic structures that can be visualized and monitored with some specific ROS tools. At last but not least, it is also important to be aware of the ROS package repositories available, providing cross-package libraries that empower a collaborative development between researchers.

ROS is still not the perfect solution in robotics, this infrastructure presents many advantages compared to other well-known frameworks, providing all the tools to perform the whole planning, reasoning, perception, or control structure required to build large robotic systems. In particular, ROS presents accurate modules for this work implementation and, in turn, will grant the empowering of all the research around the current study.

3.1.2 OpenCV Library

OpenCV is an Open Source Computer Vision Library, originally introduced by Intel for image and video analysis. Its major change took place in 2009, 10 years later from its original launch, with the transition to a C++ interface. It was a remarkable advancement for vision research due to this vision knowledge dissemination, where many programmers and researchers were encouraged to share their programming functions and optimized code, [29].

Due to the OpenCV expansion, it was crucial to find a certain interoperable programming structure, in common for all, starting with a proper namespace (i.e., cv::); It mainly works with MAT objects and each image is created as a cv::MAT, a structure holding a matrix of pixel values where each cell automatically points to an allocated memory block containing all the image data.

Since there are several algorithms and image processing techniques available in OpenCV

it is very useful to analyse the object in study through a histogram to visualize the whole picture's colors distribution through a simple graph. This image analysis provides good information about its range of intensity levels, such as brightness (higher pixels' intensity in a certain range of values), contrast (with spread pixels' intensity through the whole range of values), or even to evaluate the picture's quality with a histogram equalization (a flat range of intensity levels).

Beyond this, in image processing, it is also possible to enhance some details of interest, remove outlier noise, detect lines and corners, soften images; by combining several techniques such as mean shift algorithm, erosion, dilation, blur and apply different filters like the Laplacian, Gaussian and Sobel. However, it is important to notice that sometimes that kind of operations can affect crucial details or even discard them from the pictures like brain tumours, for instance, in medical issues.

Furthermore, the OpenCV affords a diverse set of solutions for image manipulation, making use of image descriptive files holding its most significant features, allowing to visually track some points of interest by analysing each frame of an images' sequence (video) or even make a foreground extraction by updating the picture's background with the pixels that, on average, match with the other frames. The library also enables a 3D reconstruction with the use of, at least, two cameras by matching each 2D plane projections, relating a certain image point of one camera with the corresponding line on the other's view.

In conclusion, OpenCV library affirms its position in computer vision research, not only for being an open-source tool easily accessed by students, scholars and researchers but also due to the numerous shared algorithms for image processing. Therefore, OpenCV library will be an accurate/suitable solution for some of the imagery manipulation during this implementation and an important tool to improve the knowledge growth around this science.

CHAPTER 4

Proposed Framework

The integration of new technologies and autonomous systems for precision agriculture is one of this proposed model's main goals. Particularly, autonomous drones are meant to overfly and analyse several properties and nourishment levels of the crops in order to maintain the farmers aware of their farming state of production, close to real time. Accordingly, this project is divided into three distinct parts: autonomous navigation, bi-dimensional processing and 3D analysis.

4.1 Model Overview

The UAV starts the whole process with a zigzag scanning over the crops (Figure 4.1). During this stage, a multispectral camera captures several images of the farm field that will be the basis to compute spectral known indices important for the assertion of the crops' growth and health.

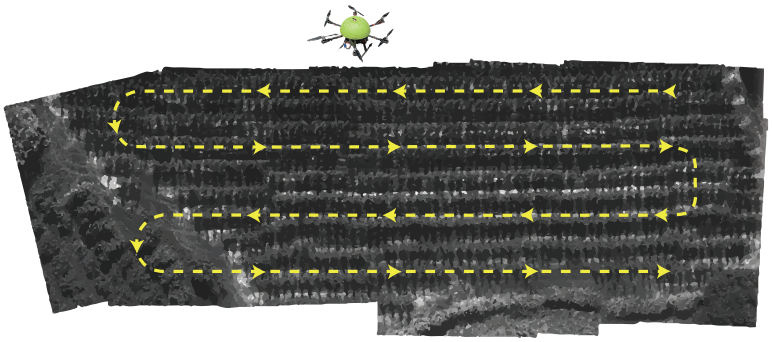


Figure 4.1: Aerial vehicle's navigation plan.

The proposed framework will also engage the whole field mapping with a terrain

classification, based on all the information acquired from the images captured by the multispectral camera. This underlying feature will be a main key to enhance an autonomous robots cooperation, regardless their type of terrain. For instance, giving aerial cooperating vehicles the ability to plan paths avoiding possible treetops during a flight, or even to enlarge the terrestrial robot's limited vision to a privileged mapped Field of View (FOV) of an aerial robot.

This particular document is focused on 2D imagery acquisition: the full required preprocessing, such as the camera calibration, the datasets preparation and the software analysis of the crops in study, as illustrated in figure 4.2.

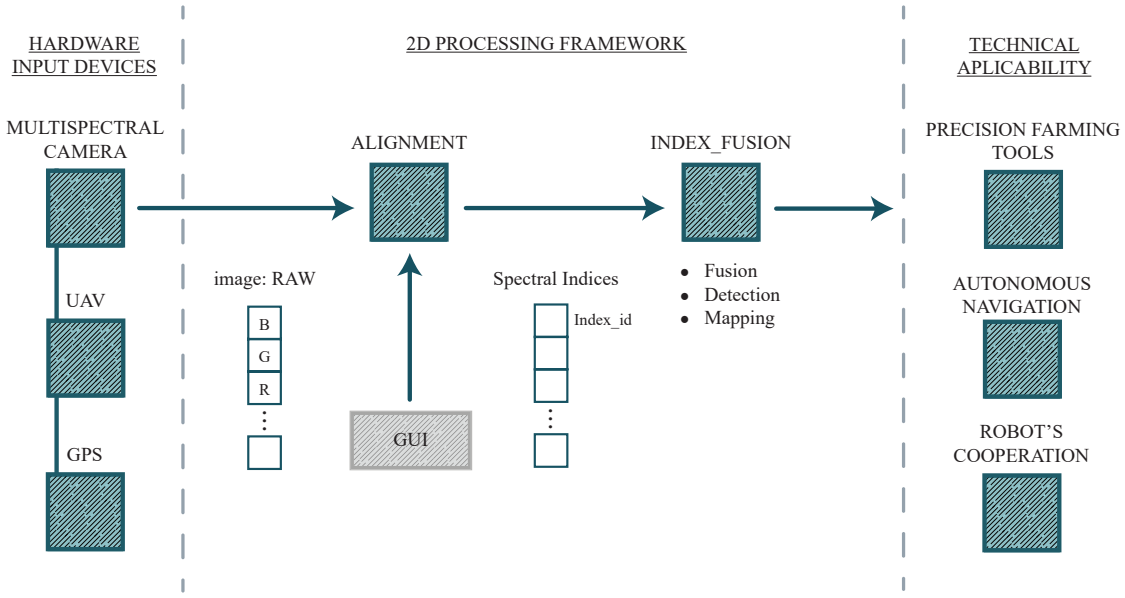


Figure 4.2: 2D Processing general work proposed structure.

After imagery calibration and alignment corrections, several scene image bands will be fused and combined, resulting some spectral indices for precision agriculture, chapter 2.2.2.1. Therefore, much information can be induced about the fields' state of production or nutritional levels.

It is also relevant to mention the ability to use spectral indices on multispectral devices to capture specific ground properties. After that, it is possible to draw some conclusions about terrain classification. This is extremely useful around the autonomous systems' field of study, where robots can cooperate with each other in several different ways.

In this case, mapping the agricultural field will, not only help the UAV self-driving control, but also cooperate with other aerial vehicles, avoiding collisions with obstacles like trees during a flight and enlarging a terrestrial robot's limited vision.

Therefore, the aerial robot's privileged field of view allows to capture an imagery dataset and process information capable of distinguishing water from vegetation, rocky terrains and sandy soils.

This terrain classification may become decisive for a robot to choose a certain route

instead of another, avoiding the unmanned vehicle to spend resources exploring an area not suitable to its characteristics and discard blocked routes in time.

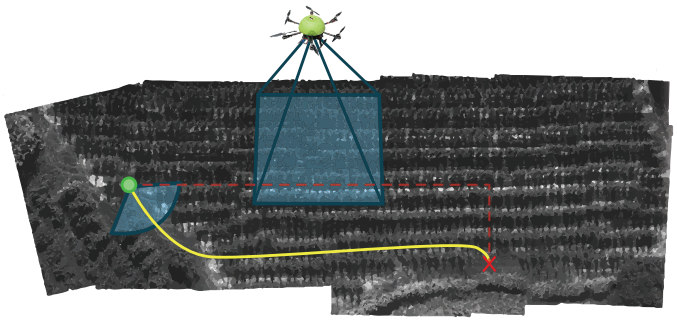


Figure 4.3: Cooperation between autonomous systems; Terrestrial robot (green dot), goal point location (red cross), blocked route (dashed line), the chosen route path (yellow line), robots' FOVs (blue areas).

Figure 4.3 presents an example of a cooperation between the UAV and a terrestrial robot (marked with the green dot). The blue areas illustrate the respective vehicles' fields of view where the aerial sight is much wider than the terrestrial perspective.

The UAV will then be able to stitch all the captured images into a map and help the terrestrial robot deciding the best possible route (yellow line) to its destination (marked with the red cross). This cooperation would avoid blocked routes (e.g. red dashed line), saving the terrestrial vehicle resources and making it faster.

In this framework, the resulting images' spectral values need to be previously parametrized and, then, mapped into several grid map's occupancy layers, representing different types of terrain, figure 4.4.

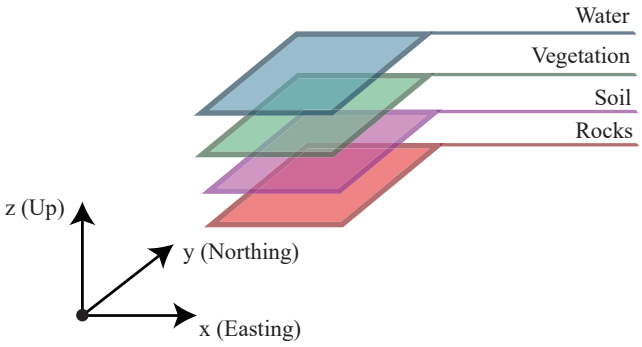


Figure 4.4: Orthophoto georeferenced layers classifying the terrains.

4.2 Hardware Infrastructures

4.2.1 Unmanned Aerical Vehicle

The aerial vehicle in use (see figure 4.5) is a six-rotor UAV, capable of landing softly, even in unexpected conditions such as a rotor failure during a flight. Under the mentioned

circumstance, despite losing it's orientation control (i.e. yaw), it is still able to auto-stabilize and land safely.

Figure 4.5: Unmanned aerial vehicle infrastructure.

Some components were added to the UAV:

- Pixhawk 2: responsible for the low-level operation's control. It is directly connected to a GPS sensor (named here+) and an IMU for positioning and orientation information. It is powered by a battery embedded with the infrastructure and communicates with a computer via MAVlink protocol. It is also connected to an UHF receiver for motors control;
- PC: a computer responsible for the software operations in computer vision and communicates via Ethernet or Wi-fi;
- ARRIS ZHAOYUN Gimbal: It is mounted under the UAV and it is where the RedEdge-M camera is placed in order to keep it stabilized and dampen flight vibrations.

The diagram in figure 4.6 illustrates the hardware layers' communication with the UAV:

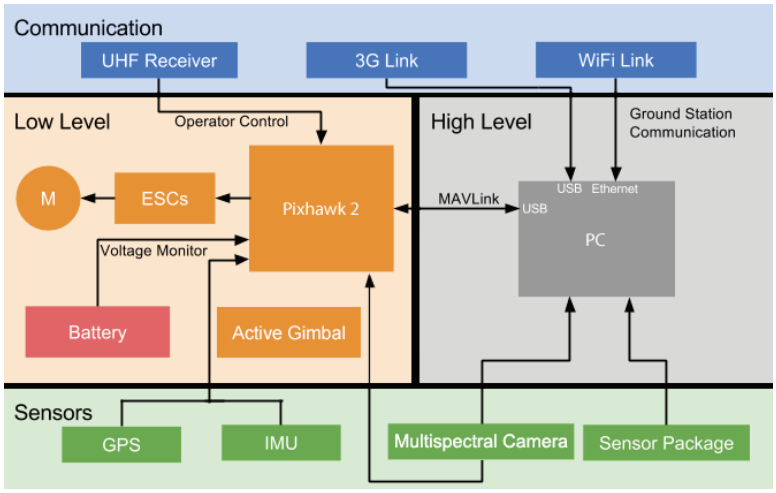


Figure 4.6: UAV hardware layer's communication's achictecture.

4.2.2 Micasense Multispectral Camera

The imagery dataset is captured by a Micasense's Rededge-M camera. The multispectral sensor has 5 different lenses, each one absorbs a distinct colour band.

This particular sensor captures the Blue (1), Green (2) and Red (3) bands from the visible spectrum, as well as the NIR (4) and Red-Edge (RE) (5) spectral bands (Figure 4.7 and Table 4.1).

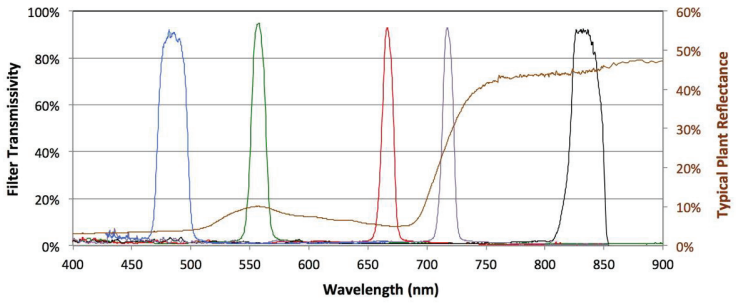


Figure 4.7: Micasense spectrum transitivity curves of the colour bands.

Band Name	Band Number	Center Wavelength (nm)	Bandwidth FWHM (nm)
Blue	1	475	20
Green	2	560	20
Red	3	668	10
Red Edge	5	717	10
NIR	4	840	40

Table 4.1: Center wavelength and bandwidth of each Micasense colour band.

This camera kit also integrates a RedEdge Downwelling Light Sensor (DLS) and a MicaSense's Calibrated Reflectance Panel (CRP). They are meant to cooperate with each other in order to make a post-processing reflectance calibration on the illumination changes during the flight.

The DLS is a 5-band incident light sensor placed on top of the aircraft, pointing forward to the flight's direction. The vehicle must be as parallel as possible to the horizon (roll and pitch angles approximately zero), maintaining a clear view of the sky during the mission. It is responsible to capture and store the amount of light information embedded in each TIFF image metadata.

Although the continuous improvements around the RedEdge DLS module, it is still not a fully independent sensor and, for this reason it is recommended to co-work with a calibration panel. Therefore, it is important to capture an image of the panel before and

after each mission, to ensure the camera and the DLS precise calibrations for shadows, reflection or light adjustments.

Micasense's RedEdge-M comes with a full bunch of integration options such as Ethernet and Wi-fi (HTTP API) or Serial (MAVLink Protocol) communication, Real Time Kinematic (RTK), and Pulse Width Modulation/General-Purpose Input/Output (PWM/GPIO) trigger, [30]. It should also be integrated with a GPS module in order to receive each image location with the GPS data (latitude, longitude, altitude, date/time) and attitude data (pitch, roll and yaw angles).

A here+ GPS Receiver is used, following the default camera integration proposed by Micasense which includes both GPS and DLS sensors (Figure 4.8).

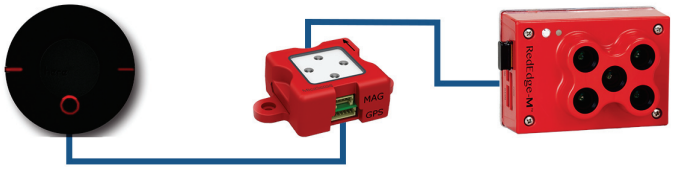


Figure 4.8: Micasense integration between Rededge-M, GPS module and DLS sensor.

The RedEdge-M sensor is ideal for the proposed project because of its lightness and dimension. It only weights 150 g and its size (12.1 cm x 6.6 cm x 4.6 cm) is also accurate for the mission, since the camera fits to the gimbal (Figure 4.9 (b)) incorporated under the UAV. The gimbal will be very important not only for imagery stabilization, but also to ensure a Nadir direction (camera directly pointing to the horizontal coordinates of the ground surface, even over sloped terrains).

Every capture is automatically stored in an SD Card, and the Raw image format can be specified along with several other specifications through the camera's web page on Wi-Fi or Ethernet mode. Each capture results in 5 different DNG (12-Bit) or TIFF (16-Bit) files according to the setting option. A 12-bit file format only means 1.8 MB per image whereas a 16-bit represents 2.5 MB for each of the 5 images, even though, the second format is still more recommended due to its higher software compatibility.

Besides, all the information is stored along with each file's metadata, which is possible to approach with any Application Programming Interface (API) like *exiftool*, for example. Micasense imagery has a 1280×960 px resolution, with an aspect ratio of 4:3. According to the Rededge-M documentation [31], the sensor lens have an Horizontal Field of View (HFOV) of 47.2^o, with a focal length of 5.5 mm and it is registered that the sensor has a Ground Sample Distance of 8.2 cm/px at a 120 m (400ft) altitude Above Ground Level (AGL).

a

b

Figure 4.9: UAV infrastructure: (a) Model infrastructure of the Micasense camera integration with the UAV. (b) ARRIS ZHAOYUN pro 3-Axis Brushless Gimbal 32bit Version.

Illustration 4.9 refers to the UAV infrastructure integration between the modules required to grant the mission performance. It is possible to observe the Rededge-M placed at the gimbal: its base is horizontally integrated with the UAV, which parallelism is granted by the yellow bubble level placed over the gimbal. On top of the aircraft, there is an opened hole at the UAV protection helmet to allow the solar rays reception by the DLS sensor. Therefore, it is also possible to observe that the integrated modules do not interfere with the UAV ergonomics and well functioning.

The development of new solutions with cameras and image processing always involve certain methodologies and imagery preparation. Before developing the framework, it was critical to be aware of the required calibration procedures, output file types and metadata formats presented by the images captured by the sensor involved (i.e., Rededge-M camera).

4.2.3 Imagery Type and Metadata

In computer vision, imagery types and formats are keypoints that need to be clearly interconnected to ensure an accurate post-processing. In this case, the image formats from the Micasense sensor are .TIFF files, which is an extension that uses a lossless compression. Images can be edited and re-saved without losing any or almost any quality.

The Micasense images metadata follows a certain structure in common: physical information about the file size, source, type, resolution, compression; and hardware specifications about the actual sensor in use, such as the focal length size, hyperfocal distance, the sensor's model or field of view value A.1.

Furthermore, the GPS sensor provides the date/time, attitude, positioning and orientation data of the image capturing moment. It also brings DLS information, where shadows or weather conditions influence specific Micasense sensor's physics variables. The earth surface information is obtained by the solar radiance reflected on the ground, without a physical contact. This data is analysed according to the irradiance (E), radiance (L) and reflectance (ρ), previously mentioned in 2.2.2.

4.2.4 Calibration Procedure

Despite the RedEdge camera capturing raw images from the fields in study, it is important to clarify that the raw pixel values from a picture do not entirely correspond to the absolute spectral radiance or the reflectance. Those values need to be previously computed for the procedures in precision agriculture, [33]. Therefore, it is required a pre-calibration of each image before it's usage.

4.2.4.1 Radiance Conversion

The Micasense radiometric calibration model is prepared to correct imagery according to the sensor properties, such as RedEdge-M sensitivity, black-level and gain, as well as to exposure settings and lens vignette effects.

Besides, calibration coefficients are scaled to only process input values of 1.0 max. Then, both the pixel and black-level values must be previously normalized by the image bit-depth (divided by 2^N, for N-bit images), in order to fit into a [0..1] range, [34].

Sensor characteristic	Parameters	Metadata related tag name
Normalized raw pixel value	p	-
Normalized black-level value	p_{BL}	EXIF:BlackLevel
Radiometric sensitivity	a1, a2, a3	XMP: RadiometricCalibration
Vignette polynomial function for pixel location (x, y)	V(x,y)	-
Image exposure time	t_e	EXIF:ExposureTime
Sensor gain setting	g	EXIF:ISOSpeed
Pixel position (x, y) ↔ (column, row)	x, y	XMP: RadiometricCalibration
Spectral radiance (SI: W/m2/sr/nm)	L	-
Pixel distance from the vignette center	r	-
Vignette center	c_x, c_y	XMP:VignettingCenter
Correction factor	k	XMP:VignettingPolynomial
Original intensity of a pixel	I	-
Corrected intensity of a pixel	$I_{corrected}$	-

Table 4.2: Micasense radiometric calibration model.

As stated in table 4.2, some of the parameters are directly and indirectly related to values from the metadata file. The indirectly related values, like the black-level (p_{BL}), gain (g) or correction factor (k), that result from variables defined at the file. However, their not determined within the same file and, for that reason, must be computed:

$$p_{BL} = \frac{BlackLevel}{65536} \tag{4.1}$$

Equation 4.1: Black-level pixel value normalized according to 16-bit images.

$$g = \frac{ISOSpeed}{100} \tag{4.2}$$

Equation 4.2: Gain's value between [0..1] from the ISOSpeed, with a base ISO of 100.0.

$$r = \sqrt{(x - cx)^2 + (y - cy)^2} \tag{4.3}$$

$$k = 1 + k_0 \times r + k_1 \times r^2 + k_2 \times r^3 + k_3 \times r^4 + k_4 \times r^5 + k_5 \times r^6 \tag{4.4}$$

.

Equation 4.4: Polynomial equation due to the computed distance from the image centre (r), 4.3

The camera manufacturer's workflow to convert a raw image values (i.e. pixels values) into radiance is as follows, [34]:

1. Accounting the black-level corrections → compute the dark pixels offset originated by a small amount of random charge generation in each pixel, a common factor for semiconductor imaging devices;

2. Apply the imagery-level effects → image compensation according to the radiometric calibration coefficients;

3. Compensate the lens vignette effect → an optical chain effect of darkening the corners of an image compared to its centre, which is also known in photography by the "light fall-off";

$$V(x, y) = \frac{1}{k} \tag{4.5}$$

Equation 4.5: Vignette polynomial function for each (x, y) pixel location.

4. Corrections due to exposure and gain → correcting the image according to the exposure time and gain settings;

5. Convert the imagery values according to the radiance system unit ($W/m^2/nm/sr$) → in respect to the exposure time, gain and radiometric coefficients compensation.

After imagery compensation over the sensor characteristics, results an absolute radiance image describing the light read by the sensor.

4.2.4.2 Radiance to Reflectance Conversion

Even so, radiance still needs to be converted into reflectance considering that, for vegetation indices or ground features' mapping purposes, this is the most important characteristic, since each material has its own reflectance spectral signature. For instance, vegetation reflectance will mostly reverberate over the green band within the visible spectrum. If it is healthy vegetation than the major reflectance will be in the NIR band. Snow, for example, strongly reflects within the visible spectrum; beyond that, it will decrease on the NIR band, until it nearly stops reflecting. See illustration 4.10.

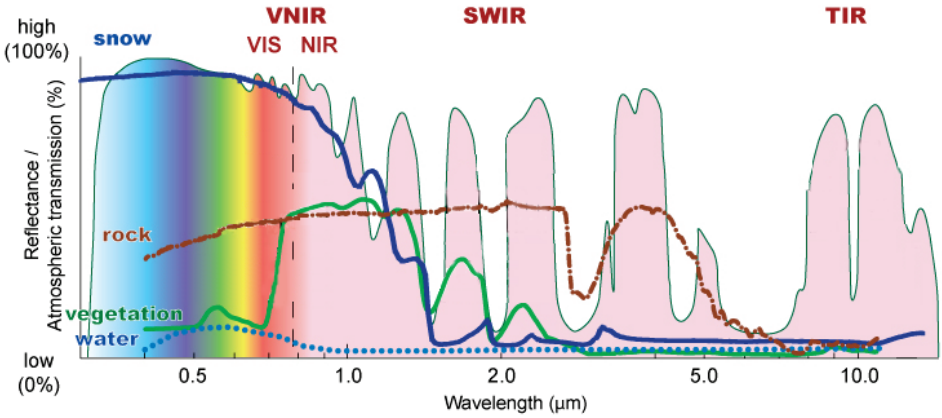


Figure 4.10: Reflectance curves of snow, vegetation, water, and rock; adapted from [35].

Therefore, this reflected light percentage, due to the total emitted light, needs to be computed. For that reason, the calibration panel will be a very useful tool.

The Micasense CRP has its own defined reflectance values for each of the 5 bands. In this ROS proposed framework, those values will be saved into a YAML file with the following structure, with the calibration panel values in use:

Blue: 0.54

Green: 0.54

Red: 0.53

Rededge: 0.52

NIR: 0.49

Accordingly, for each image calibration and conversion from radiance to reflectance, it is used a *radiance-to-reflectance* factor (F_i) determined for each specific mission. A calibration panel's picture is required at the beginning and at the end of a flight making sure there are no shadows covering the panel and the image is not saturated, as illustrated in 4.11.

Figure 4.11: Calibration panel captured by the RedEdge-M NIR sensor in use.

Since we have the real panel reflectance values, it is possible to calculate the mean value of the square panel from the radiance images taken to the CRP (defined by the red square in figure 4.11) and calculate the *radiance-to-reflectance* factor, as follows:

$$F_i = \frac{\rho_i}{avg(L_i)} \tag{4.6}$$

Equation 4.6: Radiance to reflectance factor for each *ith* band calibration.

Equation 4.6 is defined by:

- $F_i \rightarrow$ Radiance to reflectance factor for band i;
- $\rho_i \rightarrow$ Average reflectance value of the panel provided by MicaSense for band i (defined in the YAML file);
- $avg(L_i) \rightarrow$ Average of the pixels radiance intensity within the square panel computed for each band i, from the radiance image (average within the red square contour in figure 4.11)).

In essence, this radiance/reflectance factor is the key to convert every radiance image pixel to the respective reflectance value during that mission, since they might be under the same lightning conditions.

$$R_i(x,y) = L_i(x,y) \times F_i \tag{4.7}$$

Equation 4.7: Reflectance pixel value at a (x,y) position according to the respective radiance and the *ith* band radiance/reflectance factor.

Furthermore, if multiple panels are captured, a mission with k CRPs taken could be described into a linear or even a quadratic interpolation of the captured panel factors according to time.

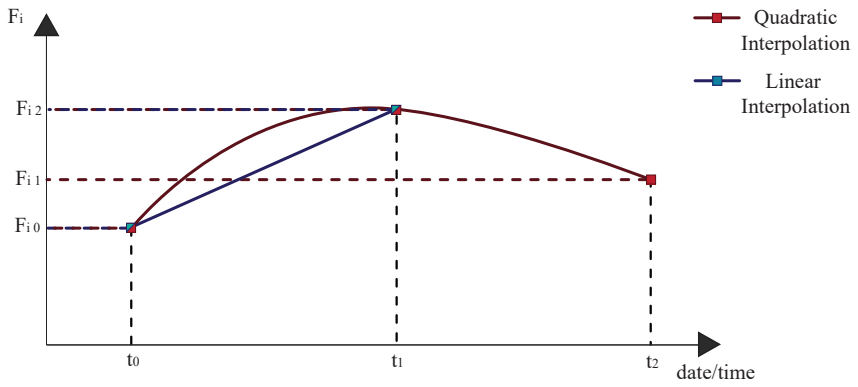


Figure 4.12: Radiometric calibration with multi-panels integration.

In illustration 4.12, the blue linear interpolation stands for an example of a mission with only two captures of the calibration panel, where t_0 is the beginning image date/time and t_1 is the ending image date/time. On the other hand, the red quadratic line represents an example of a mission divided into two parts, in this case, with an intermediate stop where t_0 is the beginning date/time capture, t_1 is the intermediate capture and t_2 is the final date/time capture.

$$\begin{bmatrix} 1 & t_0 & {t_0}^2 & \dots & {t_0}^k \\ 1 & t_1 & {t_1}^2 & & \\ 1 & t_2 & {t_2}^2 & & \\ & \vdots & & \ddots & \\ 1 & t_k & & & {t_k}^k \end{bmatrix} \cdot \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \vdots \\ q_k \end{bmatrix} = \begin{bmatrix} F_{i0} \\ F_{i1} \\ F_{i2} \\ \vdots \\ F_{ik} \end{bmatrix} \rightarrow f_i(t) = q_0 + q_1 \cdot t + q_2 \cdot t^2 + \dots + q_k \cdot t^k \quad (4.8)$$

Equation 4.8: Time-varying model $f_i(t)$ describing the reflectance factor interpolation across time to each *ith* band.

From equation 4.8, with fixed values for the reflectance factor of the panel (F_{ik}) and the date/time stamp (t_k) within the metadata file for each image, it is possible to calculate the coefficients $q_{0..k}$ and therefore find the equation $f_i(t)$.

Since every image's metadata stores the captured moment date/time stamp (t), the resultant reflectance factor to be applied should be more accurate according to the determined time-varying model $f_i(t)$.

However, the current ROS framework does not use the multi-panel corrections despite its precision, once it is an algorithm that implies post-processing methods, which means that it is not a real-time system. For that reason, the current framework only applies to imagery calibrations according to the initial CRP capture since, currently, this framework is specifically developed for real-time missions.

4.2.5 Lenses Alignments

Aside from imagery corrections, there is another important constraint about the Micasense relative provision between the sensor lenses (as illustrated in 6.2, section 6.3). These distances, among with the lenses, result in misalignments between the 5 band images taken from the same capture.

Notwithstanding, spectral indices such as NDVI or RDVI must be calculated between images that are as overlapped as possible, to assure that each object positions, within the captures, are also correlated.

In order to do this, the first step goes through deciding the best band to make the alignments with. According to the Micasense lenses arrangement, the RE lens central position should make it a good candidate because of its small distances to the other lenses.

Nevertheless, performance is one of the main algorithm's decision factors; it is important to find a band that implies less alignment processing. Since RE is not that frequently used to calculate agricultural spectral indices, which means that choosing this lens would imply as much alignments as the number of required bands to calculate the index, per image.

Overall, in a mission, this would mean $A_N = B_N \times D_N$, where A_N is the number of to do alignments, B_N is the number of required bands and D_N is the dataset total images. Then, calculating for instance NDVI (requiring the red and NIR bands), for a dataset of 100 images, would end in a total of 200 alignments during that mission.

Consequently, to avoid so much processing, it was required to choose an alignment band more frequently used within the spectral indices, such as the Red band. If so, this would imply one less alignment per image for indices where the chosen alignment band was also in use, that is $A_N = (B_N - 1) \times D_N$. For this reason, the Red band was the alignment band chosen for the ROS framework in development. Besides, it was implemented a specific aligning method capable of covering those misalignments, as detailed in section 5.5.

CHAPTER 5

Framework Implementation

5.1 ROS-based Architecture

The developed ROS-based framework is essentially divided into two main ROS nodes:

- Alignment (calibration included);
- Index fusion.

The beginning of every RedEdge-M mission has to go through a pre-camera calibration according to two distinct parameters: reflectance and lens' distortion. The panel calibration function of the alignment node implements an initial calibration for each of the five bands' images.

Based on the official Micasense Python calibration's script [33], the calibration methods were totally converted into a new C++ open-source library, used by the proposed framework. This library intends to be available to any developer who needs the C++ Micasense correction methods, with the advantage of being available in ROS systems (*micasense_ROS.lib*).

After the calibration procedure, the images are aligned and the index fusion computation can be performed. The final results from the index fusion node may then have several purposes, in this case, the framework will be specifically used to support the autonomous vehicle's navigation or even to strengthen farmers with a detailed view over their crops through a map.

Figure 5.1 represents and illustrates a general diagram of the implemented infrastructure:

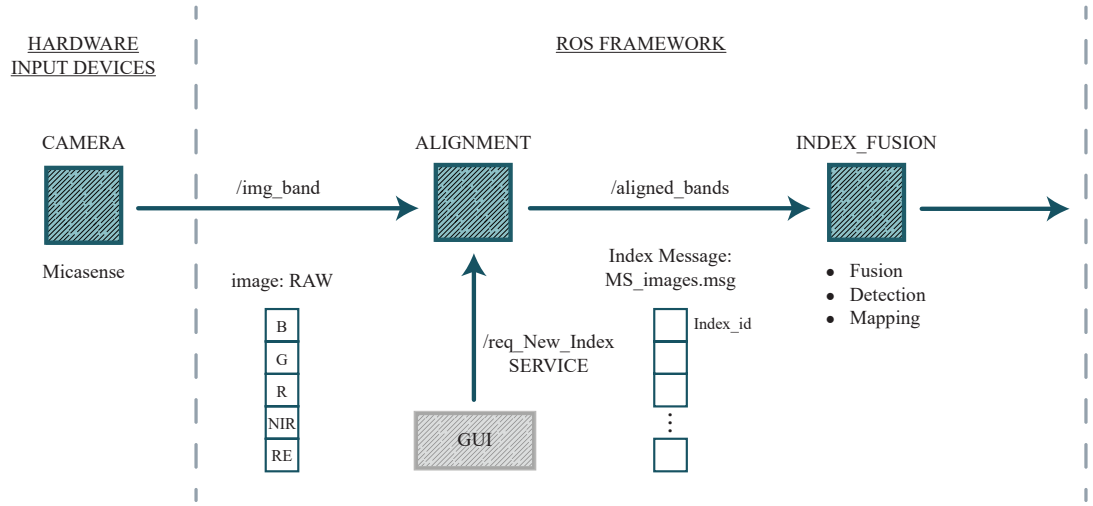


Figure 5.1: ROS implemented infrastructure's diagram.

Although the implemented ROS-based framework was designed, knowing that the input device would be a RedEdge-M camera, it can be fully integrated with any other imagery input device, from the simplest RGB cameras to complex hardware systems like satellites.

The framework's deploy is triggered by a service (*/index_request*) that enables/disables an intended index computation by a string, indicating the requested index name and the new activation state linked by an underscore (e.g. NDVI_on, NDVI_off, RDVI_on). However, it could also be integrated with any Graphical User Interface (GUI), capable of injecting any requests and even return useful data to the end-user through a more appealing platform.

5.1.1 Service Triggering Operations

As previously stated, the trigger's routine is implemented as a ROS service (*Req_New_Index.srv*) and it is prompted as a */req_New_Index* topic followed by a string such as: "*(requested index name)_(new state)*".

The requested index name is ready to receive any of the following data:

- NDVI → Equation 2.1;
- RDVI → Equation 2.3;
- ENDVI → Equation 2.2;
- MSR → Simple ratio equation based on 2.4;
- MSAVI → Equation 2.6;
- ALL;
- PANEL.

The former available index names trigger the respective requested index, where the "ALL" instruction triggers all the available indices to on/off.

As the name itself suggests, the new state is ready to receive an "on" or "off" instruction (in order to enable or disable an index respectively), as well as an "ALL" instruction that includes the above mentioned options.

Moreover, the "PANEL" instruction is used during online missions to call method responsible for setting the panel calibration values dependent variables (further detailed in chapter 5.2).

When the *Req_New_Index.srv* service is triggered, it prompts the *user_requested_index* method, which is responsible for managing user requests and keeping both *Requested_Indexes[]* and *Required_Bands[]* arrays updated:

Requested_Indexes[] → [NDVI, ENDVI, ...]

Required_Bands[] → [B | G | R | NIR | RE]

Whereas the *Requested_Indexes[]* array adds the new index names, the *Required_Band[]* is a band's counter, incrementing the corresponding array index (required to calculate the requested spectral index). For example, if only the NDVI (using the Red and NIR bands) and the ENDVI (using the Blue, Green and NIR bands) are activated, then the *Required_Bands[]* will be [B: 1 | G: 1 | R: 1 | NIR: 2 | RE: 0].

5.1.2 ROS Messages

The framework infrastructure node's communication is made by advertising or subscribing messages sent by/to the respective topics. The same ROS message type is sent to communicate through the */img_band* and */aligned_bands* topics, which is defined in the *MS_images.msg* file:

```
# Constant Definition for active operating mode
string[] Indice_Name
int8 Img_Index
sensor_msgs/Image[] Images
sensor_msgs/NavSatFix GPS
float64 Panel_Alt
```

The */img_band* topic makes use of the *Img_Index* variable, which is basically a capturing counter and it is also composed by two of the ROS sensor message (*sensor_msgs*) types: Image [41] and NavSatFix [42]. The former type is used as an array of images, storing the five bands belonging to each capture; whereas the latter, NavSatFix message type, is defined by GPS information, storing the vehicle's location with the respective coordinates (latitude, longitude and altitude), and the *position_covariance*[0] float value that retrieves the UAV orientation (i.e. yaw).

In contrast, the */aligned_bands* topic fully uses the five message variables, including the *Indice_Name* as the requested index fusion identifier and the *Panel_Alt* as the captured panel's altitude with a referenced positioning height according to the sea level.

Moreover, although the */img_band* topic sends the five raw images through the *Images* vector, the */aligned_bands* only send the requested index fusion's corrected images.

5.1.3 RedEdge-M Communication Node

RedEdge-M camera data can be retrieved by serial communication (based on the MAVLink protocol) or HTTP requests. The proposed framework uses an HTTP communication, via either Wi-fi (802.11) access point or Ethernet connector.

By default, the multispectral camera's communication is performed via HTTP connection to port 80 at the IP, address according to the configured accessing method:

- Wi-Fi: 192.168.10.254
- Ethernet: 192.168.1.83

Each operation's command can be triggered either by GET or POST requests, resulting in a JSON object as response. For instance, the URL:*/gps* request could result in a JSON message, such as:

```
{
"latitude" : 0.1363053,
"longitude" : -2.399923234,
"altitude" : 332.503,
"vel_n" : 0.73,
"vel_e" : 1.2,
"vel_d" : 0.49,
"p_acc" : 2.456,
"v_acc" : 1.54,
"fix3d" : true,
"utc_time" : "2014-10-08T20:27:23.321Z"
}
```

Apart from collecting data, it is also possible to trigger a new capture from an HTTP request or even attach a different GPS location instead of the one provided by the external sensor. Even so, the waypoints to trigger are previously defined before each mission, and this API is essentially used to read the sensor's captured images during the flight.

Regarding the performance time, Wi-fi connection is extra hardware components' free (due to its wireless connection), however, it may result in slower communications and signal losses. On the other hand, Ethernet wired connection is much faster but it requires more hardware components mounted on the aircraft, decreasing the UAV flight-time. Nevertheless, both HTTP connections will be tested for the developed framework.

5.2 Online and Offline Missions - User Interface

Regardless the running mission type (online or offline) the user has always to choose the panel calibration's folder location. For offline missions, dataset directory is also needed.

However, if the algorithm is running online, the user only needs to make sure that the panel's capture, under the mission's presented light conditions, is previously stored in the computer (PC module, figure 4.6). Then, before starting the flight, by calling the "PANEL_on" *Req_New_Index* service's option, the user is able to choose the panel's directory folder to apply the proper correction's parameters.

On the other hand, when running offline missions, the user is automatically prompted with the mission's dataset and the panel's capture directory folders' location.

5.3 Image Metadata Extractor

The RedEdge-M images bring several useful data for this framework's implemented procedures. Hence, before saving an image file, it is important to verify if the corresponding metadata is still needed. OpenCV image's saving functions (*imwrite()*) do not attach the referencing information, unless it is previously copied and written as new tag values into the file to be created.

Exiftool, by Phil Harvey ([32]), is a C++ library interface used for this purpose. It extracts the metadata from images, storing its information and capturing sensor data. One of the first attempts to reach this data was taken from one of the collected datasets' random image, in this case, a calibration panel's NIR band picture (written in a YAML sample file with image metadata).

Rededge-M metadata is structured in *exiftool* with values, separated by line breaks, identified with the corresponding tag name, linked from its value through a colon, as exemplified bellow:

PressureAlt: "57.867588043212891"

DarkRowValue: "5059, 5083, 5056, 5040"

BandName: NIR

CentralWavelength: "840"

VignettingCenter: "617.11652352677311, 474.27859714857334"

FOV: "48.8 deg"

GPSPosition: "38 deg 39' 32.22" N, 9 deg 12' 20.57" W"

FocalLength: "5.5 mm"

However, since each information has its own structure, separated by different symbols and incorporated with different system units, it also requires an individual processing to withdraw the required information and, therefore, assign it to proper variables.

Moreover, before starting to extract the desired information, it is critical to verify if the metadata file does not bring duplicated tag names (since *Exiftool* interface is only prepared to return one value for each tag name), because this solution replaces the tagged variables by the last read values with that referred tag name. For that reason, metadata has to be previously written on, for example, a YAML file. This ensures detectable replicated tag names: if the correct values are not defined at the last referenced tag name, external processing methods are required to access the desired data.

This situation is detected twice in this work: the first one, is related to the GPS information which, in this case, is not an issue because the values match; on the other hand, the attitude information is also replicated (with the tag names: Yaw, Pitch and Roll). The latter is important to notice since the last interpreted values can be misunderstanding - they do not refer the actual UAV attitude (the desired information) but the DLS sensor's irradiance attitude. In future mapping procedures, this will be a critical spotlighted detail.

Nevertheless, during further processing, the *Exiftool*'s metadata extractor is a very simple and efficient tool.

5.4 Reflectance Conversion

5.4.1 Converting Raw Images to Radiance

The multispectral camera images sent to the *Alignment Node* are composed by their initial raw pixel values. However, the raw pixels need to be converted into radiance, according to the micasense work-flow steps described in chapter 4.2.4.1 and supported by table 4.2.

```
/********
METHOD: raw_image_to_radiance(imageRaw);
INPUTS:
imageRaw: Raw image
OUTPUTS:
radianceImage : Image with raw pixel values converted into Radiance values.
VARIABLES:
Exfitool Values (Table 4.2):
darkLevel:
black_levels:
a1, a2, a3:
exposureTime:
gain: equation 4.2
bitDepthMax: 2^16 for 16-bit images
Y: Coordinate grid across image rows -> Transposed
V: Vignette polynomial function for each (x, y) pixel location, (k in
    equation 4.4).
R: Radiometric calibration's compensated image
```

```
18 L: "Light fall-of" compensated image
19 ********/
```

1. Accounting the black-level corrections - Black-levels average:

```
1 darkLevel = average(black_levels[]);
```

2. Apply the imagery-level effects:

```
1 R = 1.0 / (1.0 + a2 * Y / exposureTime - a3 * Y)
```

3. Compensate the lens vignette effect:

```
1 L = V * R * (imageRaw - darkLevel);
```

4. Corrections due to exposure and gain:

```
1 //exposureTime -> exiftool value: ExposureTime (te in Table 4.2)
2 //gain -> equation 4.2
```

5. Convert the imagery values according to the radiance system unit ($W/m^2/nm/sr$):

```
1 radianceImage = L / (gain * exposureTime) * a1 / bitDepthMax;
```

Subsequently, from the computed radiance pixels' values it is possible to convert radiance into reflectance, according to the *radiance-to-reflectance* factor (Fi). This implementation is discriminated in the following chapter 5.4.2.

5.4.2 Panel Detection and Calibration

After converting the raw panel image to radiance, the framework also automatically detects the desired radiance square by making an homography-based cropping routine, to find the QRcode embedded along with the micasense calibration panel and the accurate square's location. The homography is made accordingly to a previously stored panel base image with both QRcode and radiance's square, as illustrated in figure 5.2.

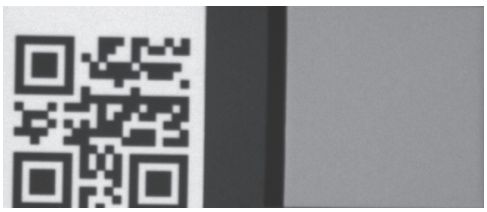


Figure 5.2: Calibration's panel base image with the QRcode and radiance square to use on automatic detection procedures.

The *panelHomography* method uses this image to find its pattern within the actual mission's calibration panel frame.

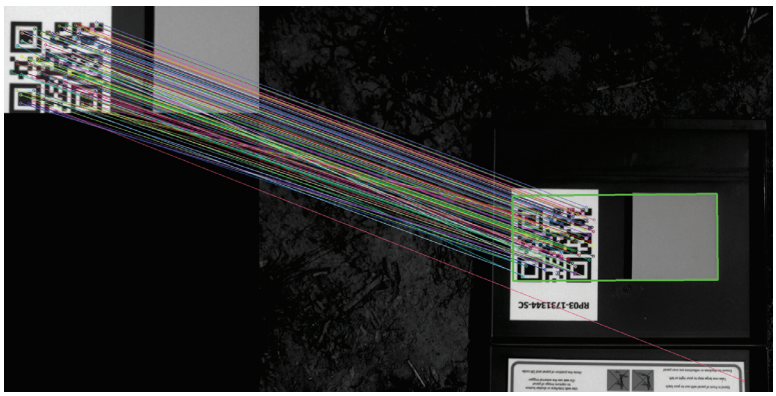


Figure 5.3: Panel Detection: example of an homography good matches result.

Furthermore, the homography detection is implemented to find the desired pattern even if compared with a rotated or inverted panel's image, as illustrated in figure 5.3.

After finding the corresponding image 5.2 location within the new panel's capture, the latter is cropped. Afterwards, the radiance square location's automatic detection is performed according to the vertical and horizontal projections' histograms, from the binary detected image.

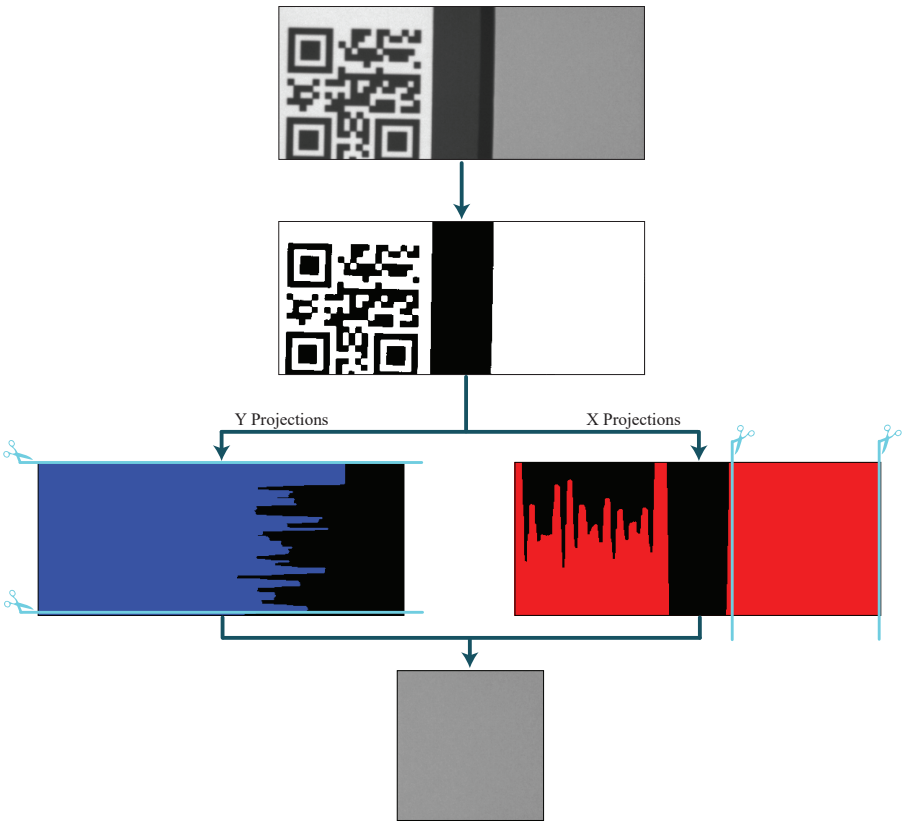


Figure 5.4: Radiance square detection diagram.

Illustration 5.4 describes the panel detection routine. The binary image is white (255) for imagery values greater than 80% of the average image intensity, and black (0) for values bellow. Hence, the binary image's horizontal and vertical projections' histograms are used to accurately segment the reflectance square.

From the resulting panel's cropped square (defined as the *panelRegion*) the average radiance's intensity is measured by computing every radiance square panel's pixels. Accordingly to the used panel's specific reflectance values (saved in the *panelCalibration.yml*) it is possible to compute the *radianceToReflectance* factor (as previously explained in chapter 4.2.4.2).

```
/********
METHOD: calib_init(Panel_rawImage) -> Initial Panel Calibration routine
INPUTS:
Panel_rawImage: Panel's raw image.

OUTPUTS:
reflectanceImage : image converted into reflectance.

VARIABLES:
path: path to the "panelCalibration.yml" file
panelReflectance: Reflectance value of the "Panel_rawImage_band" parsed from
    the YAML file
radianceToReflectance: radianceToReflectance factor (Fi in equation 4.6)
********/
Panel_radianceImage = raw_image_to_radiance(Panel_rawImage)

// Panel homography detection routine
panelRegion = panelHomography(Panel_radianceImage)

meanRadiance =  average(panelRegion)

radianceToReflectance = panelReflectance / meanRadiance;

reflectanceImage = radianceImage * radianceToReflectance;
```

The sensor's received images are easily converted from radiance to reflectance, from the product between every pixel's radiance intensity and the *radianceToReflectance* factor:

```
flightReflectanceImage = radianceImage * radiance_To_Reflectance;
```

5.4.3 RedEdge-M Lens Distortion Corrections

After the reflectance calibration, the lens distortion correction's routine, based on camera model's calibration, is applied using the C++ micasense library (figure 5.5).

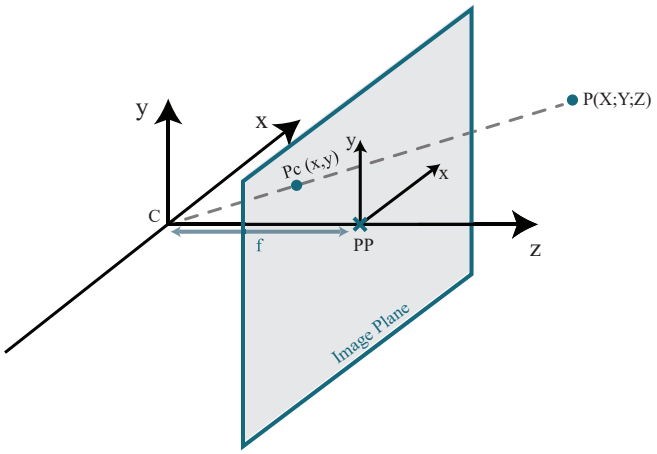


Figure 5.5: Camera Model's Calibration where: C - camera centre; PP - principal point; Pc - Point on image; P - point Pc in world coordinates; f - focal length. Equation 5.1

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \sim \begin{bmatrix} fX \\ fY \\ Z \end{bmatrix} = \begin{bmatrix} fx & 0 & cX \\ 0 & fy & cY \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{5.1}$$

Equation 5.1: Camera Model's Calibration

```
/********
METHOD: correct_lens_distortion(refl_image)
INPUTS:
refl_image: Reflectance image.

OUTPUTS:
undistortedImage : Reflectance image after lens' distortion corrections.

VARIABLES:
xPP, yPP : Principal point (Exiftool extracted parameter)
cX , cY : Camera centre.
fx, fy, focal_length_mm : Perspective focal length
dist_coeffs: Perspective distortion coeficients. (Exiftool extracted
    parameter)
FocalPlaneXResolution, FocalPlaneYResolution: Focal lane resolution (
    Exiftool extracted parameter)
********/
cX = xPP * FocalPlaneXResolution;
cY = yPP * FocalPlaneYResolution;
fx = focal_length_mm * FocalPlaneXResolution;
fy = focal_length_mm * FocalPlaneYResolution;

Camera_Matrix(0, 0) = fx;  Camera_Matrix(0, 2) = cX;
Camera_Matrix(1, 1) = fy;  Camera_Matrix(1, 2) = cY;
Camera_Matrix(2, 2) = 1.0;

```

```
new_cam_mat = cv::getOptimalNewCameraMatrix //Camera matrix based on the
    distortion coefficients.
map1, map2 = cv::initUndistortRectifyMap //Undistortion and rectification
    transformation maps.
undistortedImage = cv::remap //Geometrical maps' transformation to the image
```

This method corrects the images provided by the camera to obtain accurate results.

5.5 Image Alignment

After the lenses distortion corrections, followed by the reflectance calibration, the RedEdge-M pictures are corrected according to a misalignments' rectification (i.e., image alignment).

5.5.1 Alignment Algorithm

Micasense documentation suggests an image alignment method based on an OpenCV library example, which uses a motion model based algorithm named Enhanced Correlation Coefficient (ECC), [36].

In brief, OpenCV has four distinct motion models, illustrated in figure 5.6:

- Translation → Two scene images shifted by (x,y), (2 parameters);
- Euclidean → Two scene images shifted by (x,y) and/or rotated by an angle, (3 parameters);
- Affine → Two scene images transformed by rotation, translation, scale, and shear, (4 parameters);
- Homography → Two scene images transformed according not only to 2D effects (as the previous ones) but also with some 3D transformations (8 parameters).

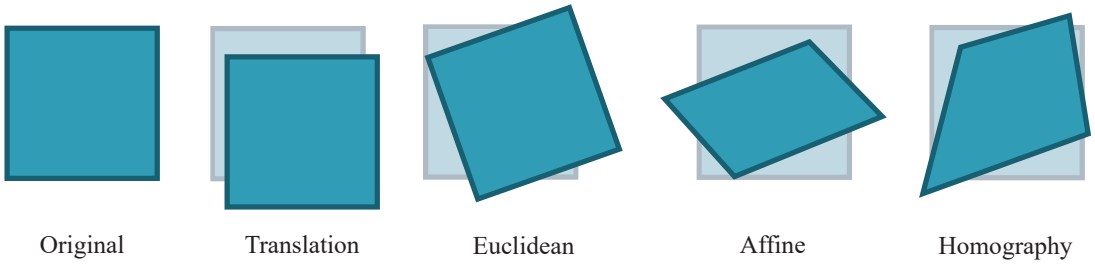


Figure 5.6: Radiometric calibration with multi-panels integration.

Furthermore, for the proposed model use case, although the agricultural provisioned datasets imply a certain altitude from the soil, and those lenses misalignments might be even small (illustration 5.7), they still have to be corrected using similar motion model transformations.

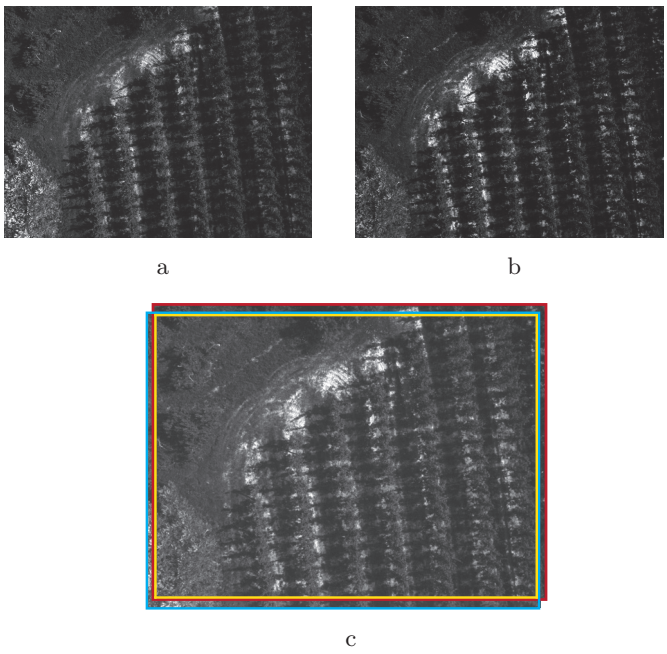


Figure 5.7: Misalignments between scene images from different lenses. The blue square is surrounding the blue band image (a), whereas the red square is the contour of the red band image (b). The yellow square represents the cropped images after transformations and illustrates the scene area in common (c).

Accordingly, the OpenCV code was tested in four distinct proposed models of the ECC alignment, [36]. Regarding the accuracy, it wasn't expected a successful alignment result using the translation or the euclidean transformations, since neither of them have enough parameters to correct lenses distortion effects.

On the other hand, both affine and homography models were good options to ensure alignments' accuracy, due to their ability to solve focusing operations, by applying adjustments not only to shift but also to slightly scale or even shear the images.

The OpenCV aligning models were then tested with two different strategical images included in one of the use case datasets of a vineyard. Whereas the first image (Image 1, figure 5.8) was expected to perform the adjustments easier (in terms of computer vision, there are more visible outstanding features that could be easily identified), the second image (Image 2, figure 5.8), was chosen to verify if the test would be able to identify the correct similarities in images with not so many distinguishable features.

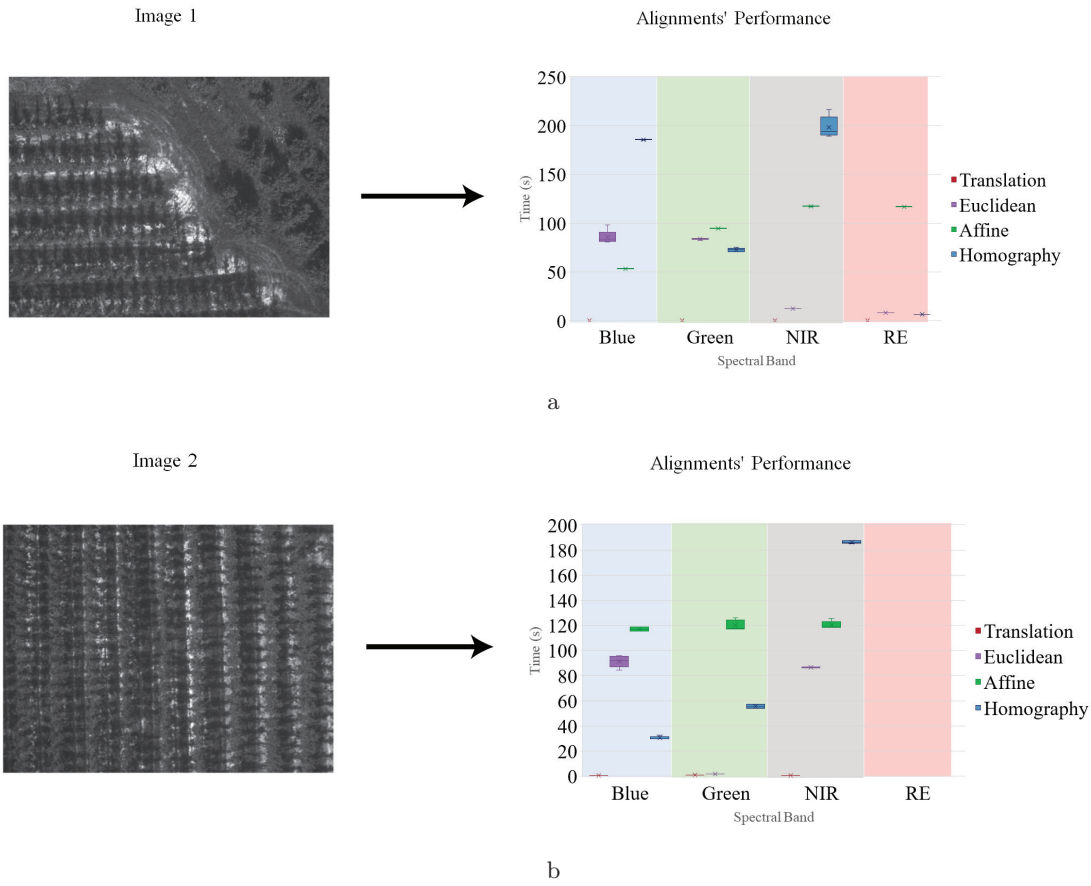


Figure 5.8: Performance test of the ECC alignment algorithm with two distinct images: (a) Image 1 - Scene image with several outstanding features. (b) Image 2 - Scene image with constant similarities.

As illustrated in figure 5.8, each OpenCV model was tested by correcting each band (blue, green, NIR and RE) according to the red band scene image. The results were not exactly the expected ones since the four models failed in, at least, one of the two main requirements: processing time and successful result.

Figure 5.8 represents the algorithm's success concerning all the band filters, tested in every OpenCV transformation model. The former image (5.8 (a)) illustrates a completely successful result, whereas the latter (5.8 (b)) shows a good alignment for blue, green and NIR spectral bands, except for the RE failed alignment.

As for processing time, despite the accuracy successful results, the translation method would be the only acceptable one for the near real-time proposed model. In average, the translation model spent 0.37s per image processing, whereas all the other methods took more than one minute to process each image.

To sum up, all accuracy, processing time and success rate factors were determinant to decide that the ECC algorithm was not suitable to the work in development.

Instead, the alignments' approach will be based on another OpenCV library method [37] developed to detect objects within an image (e.g., to find a book within a messy room,

figure 5.9). It is a fusion between a bi-dimensional features detector and a method capable of finding the homography matrix between two planes.

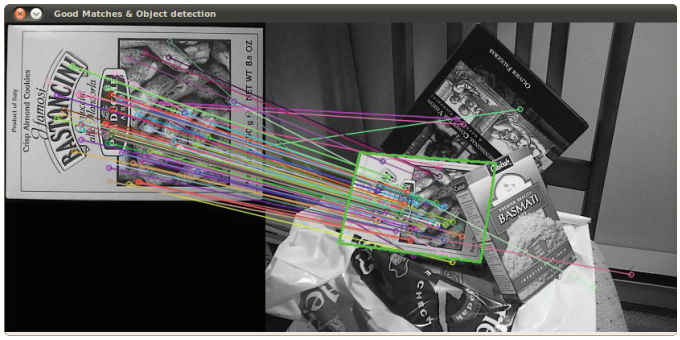


Figure 5.9: Example of the OpenCV library algorithm: Features2D + Homography, [37].

The OpenCV method uses SURF features detector and descriptor, which is one of the most robust and fast algorithms, invariant to scale and rotation. It is based on the Hessian matrix which considers the convolution of the Gaussian second order derivative of an image. However, its computation speed is essentially due to its input of integral images and Gaussian approximations using box filters, [38].

To apply to the agriculture case study, from the detected aligning images' keypoints in common (using Fast Library for Approximate Nearest Neighbors (FLANN) matcher [39]), a perspective transformation matrix (homography) is computed between both scenes. Then, the image to be aligned will be cropped from the overlapped detected corners and stretched to the same size as the red filtered.

5.5.2 Image Alignment Node

The framework's alignment node is the main responsible for converting and aligning the captured images and, subsequently forward the results by publishing them to the */aligned_bands* ROS topic.

First, the alignment node's subscriber listens to the */img_band* topic and when it receives a new raw captures' data from the camera, structured as a ROS message on the *MS_images.msg* file, the *received_img()* routine is prompted.

In particular, the *received_img()* method receives all the images from the camera module, either from the RedEdge-M or any other multispectral device.

At this stage, each capture starts to be accurately corrected according to the methods previously explained in chapter 5.4.

```
1 /********
2 METHOD: micasense_Corrections()
3 INPUTS:
4 img: Image to be corrected due to Reflectance and Lens' Distortion.
5
6 OUTPUTS:
```

```
flightUndistortedReflectance : Reflectance image after lens' distortion
    corrections.

VARIABLES:
radianceToReflectance: radianceToReflectance factor (Fi in equation 4.6)
********/
radianceImage = Micasense::raw_image_to_radiance(img); // Method from the
    new Micasense C++ library
flightReflectanceImage = radianceImage * radianceToReflectance;
flightUndistortedReflectance = Micasense::correct_lens_distortion(
    flightReflectanceImage); // Method from the new Micasense C++ library

return flightUndistortedReflectance;
```

After the multispectral imagery correction, it is possible to proceed with the lenses' alignment.

The image alignments are performed according to the previously mentioned homography method. Even so, given the images' homogeneity, the contours need to be highlighted before running the homography routine to detect similar features between the captured red band (*R_band*) and the aligning band (i.e. *blue_band*, *green_band*, *RE_band* or *NIR_band*).

Therefore, it was applied an extra Laplacian filter on the images before the homography method (*imageHomographyAlignment()*). Furthermore, in order to increase the accurate matches between images, the algorithm also discards the points outside the possible range location (i.e. distance greater than 2% of the image size). These methods significantly increased the correct number of features detected between both images, which ensures a better alignment at the end.

```
/********
METHOD: imageHomographyAlignment(aligning_band, red_band)
INPUTS:
aligning_band: Band to be corrected and aligned according to the Reference
    Band.
red_band: Reference band to align the other images with.

OUTPUTS:
aligned_band : aligned_blue, aligned_green, aligned_RE or aligned_NIR

VARIABLES:
frame1_Laplacian = Laplacian applied to aligning_band.
frame2_Laplacian = Laplacian applied to red_band.
min_dist = Minimum distance between keypoints. Details in [37].
********/
cv::Laplacian(aligning_band, frame1_Laplacian, thresh1, thresh2);
cv::Laplacian(red_band, frame2_Laplacian, thresh1, thresh2);

Surf_detector = cv::xfeatures2d::SURF::create()
keypoints_1 = aligning_band features according to Surf_detector
```

```
keypoints_2 = red_band features according to Surf_detector
SURF_extractor = cv::xfeatures2d::SURF::create()
matches = FlannBasedMatcher matches result from the images descriptors

for all i between 0 and descriptors_1.rows do
if    matches[i].distance    less than    cv::min_dist)
good_matches = matches
//Input only the points inside the 2% range:
for all good_matches between 0 & keypoints_1.size do
diff = distance_diff_percent(between    good_matches_keypoints_1    and
    good_matches_keypoints_2) //distance difference percentage
if    diff    less than    2%
add keypoints_1 (x,y) to frame_1_points
add keypoints_2 (x,y) to frame_2_points
end for
end for

H = cv::findHomography(frame_1_points, frame_2_points) //Homography Matrix
cv::warpPerspective(aligning_band, aligned_band, H) //Mode: BORDER_REFLECT

return aligned_band;
```

After the image bands, required by the user's requested indices, are aligned, a new ROS message is filled (as the *MS_images.msg*, chapter 5.1.2). This message will be published to the */aligned_bands* topic which is listened by the framework's next stage node: Index Fusion Node; further explained in chapter 5.6.

5.6 Index Fusion

According to the index fusion node, the alignment node's received message needs to be decomposed to obtain the turned on indices, the images required to calculate those requested indices and the respective GPS and IMU sensor's information.

Hence, the index fusion node is essentially composed by the following three steps:

1. Spectral indices bands' fusion;
2. Terrain classification;
3. Mapping.

Whereas the first step is directly calculated using methods developed for index fusion, the last two steps belong to another fully implemented C++ ROS library, which methods will be called and used by further index fusion node's routines.

Similarly to the previously mentioned developed *micasense.lib*, this new Layer Mapping library(*layer_mapping.lib*) will also be open-source, in order to help other developers with methods specifically implemented for imagery classification and mapping, as further explained in chapter 5.6.3.

5.6.1 Spectral Indices Computation

According to the mentioned implemented indices, in chapter 5.1.1, these bands' fusion are the key for precision agriculture or, in this particular case, for the terrain classification methods. After all, it is the spectral index fusion that brings the interesting features out from the images' reflectance values.

The index fusion node is composed by the five implemented indices' calculation routines: NDVI, ENDVI, RDVI, SR and MSAVI. These methods scan the image pixels and, for each cell, computes the index value according to the formulas presented in chapter 2.2.2.1.

Despite the fact that only these five spectral indices were implemented, the Index Fusion Node can be easily adapted to receive and calculate other new indices to the framework.

5.6.2 Terrain Classification

According to the requested indices' resultant images, the spectral values were parametrized to classify each terrain type (i.e., vegetation, water, soil and rocks).

Therefore, for each implemented index, the best possible values range was determined to the respective terrain types:

Terrain Types Range Parametrization	Vegetation	Water
NDVI	[NDVI_VEG_MIN ; NDVI_VEG_MAX]	[NDVI_WATER_MIN ; NDVI_WATER_MAX]
ENDVI	[ENDVI_VEG_MIN ; ENDVI_VEG_MAX]	[ENDVI_WATER_MIN ; ENDVI_WATER_MAX]
RDVI	[RDVI_VEG_MIN ; RDVI_VEG_MAX]	[RDVI_WATER_MIN ; RDVI_WATER_MAX]
SR	[SR_VEG_MIN ; SR_VEG_MAX]	[SR_WATER_MIN ; SR_WATER_MAX]
MSAVI	[MSAVI_VEG_MIN ; MSAVI_VEG_MAX]	[MSAVI_WATER_MIN ; MSAVI_WATER_MAX]

Terrain Types Range Parametrization	Soil	Rocks
NDVI	[NDVI_SOILS_MIN ; NDVI_SOILS_MAX]	[NDVI_RCK_MIN ; NDVI_RCK_MAX]
ENDVI	[ENDVI_SOILS_MIN ; ENDVI_SOILS_MAX]	[ENDVI_RCK_MIN ; ENDVI_RCK_MAX]
RDVI	[RDVI_SOILS_MIN ; RDVI_SOILS_MAX]	[RDVI_RCK_MIN ; RDVI_RCK_MAX]
SR	[SR_SOILS_MIN ; SR_SOILS_MAX]	[SR_RCK_MIN ; SR_RCK_MAX]
MSAVI	[MSAVI_SOILS_MIN ; MSAVI_SOILS_MAX]	[MSAVI_RCK_MIN ; MSAVI_RCK_MAX]

Table 5.1: Terrain types parametrization according to each multispectral index.

Finally, after processing each image index result, the frames are analysed and each pixel is classified and accurately mapped to the respective layer (GPS location), using the Layered Mapping library procedures.

5.6.3 Semantic Layer Mapping

Nowadays, grid maps are extremely used in mobile robotic mapping; there is a specific C++ Grid Map library, as a ROS interface, capable of storing distinct types of terrain features, like elevation, variance and colour ([40]). Grid maps can also have an unlimited number of layers and it is possible to convert them into other ROS message types, such as *PointCloud2*, *OccupancyGrid* and *GridCells*. For viewing purposes it is also fully integrated with the ROS' 3D visualization tool, known as *RViz*.

This framework's pipeline final step is to classify terrain features from the received frame's spectral indices and map each cell classification into grid map layers, as mentioned in chapter 4.1. Moreover, the terrain is classified according to specific parametrized values and, therefore, the resultant pixels are computed into dynamic grid map layers.

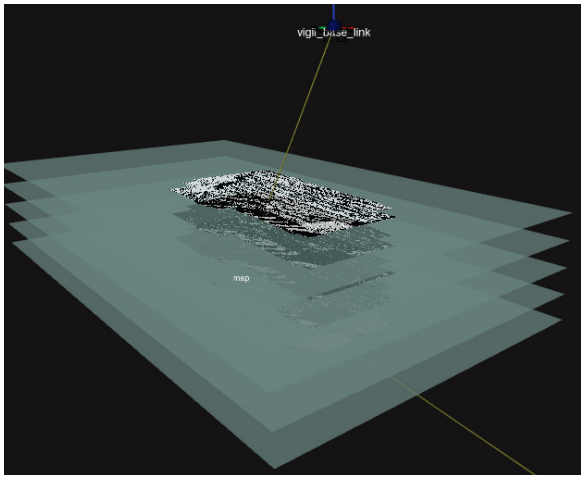


Figure 5.10: Terrain Layers viewed from the Rviz tool

In essence, this imagery mapping forms a stitched image of the full area in study. Besides, the framework, not only builds a layer generated with the respective index fusion images, but also a layer for each terrain classification, from the respective calculated index.

In fact, for this case study, there will be 5 layers for each index fusion:

- 4 terrain classification layers (i.e. vegetation, water, soils and rocks);
- 1 spectral index fusion layer (e.g. NDVI images' layer).

In other words, if only the NDVI is turned on, there will be four terrain layers: NDVI_vegetation_map, NDVI_water_map, NDVI_soils_map and NDVI_rock_map layers; with one more layer for mapping the stitched NDVI images.

Along the images classification, the grid map will be gradually filled and converted into the standard ROS message in use (i.e. nav_msgs/OccupancyGrid) and then the several layers are published to the corresponding ROS topic. Through the *RViz* tool it is possible to visualize every layer map being published, near in real-time.

This mapping procedures has many possible applications, such as in vehicle's autonomous navigation or robots cooperation tasks.

5.6.3.1 Imagery Altitude Considerations

Usually during an UAV flight, there are several variables that need to be considered. As for mapping, fluctuating flight altitude variables have to be computed. Although the camera images have a fixed sensor dimension, before mapping an image, knowing the real image's dimension is a requirement. Otherwise, the images would never be properly mapped.

The RedEdge-M datasets are composed by images with a $4:3$ proportion sized by 1280×960 px, but for performance improvement, the images are resized to 640×480 px. Considering the sensor's image size in pixel, it is possible to compute the real area dimension from the camera's field of view angle (FOV), figure 5.11.

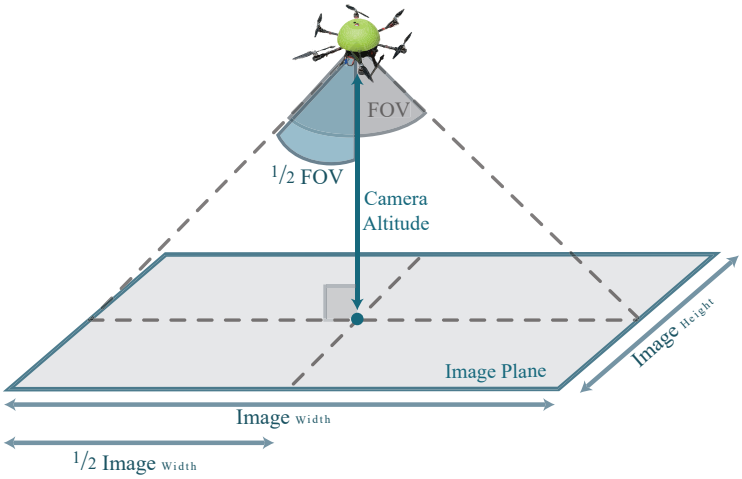


Figure 5.11: UAV pipeline to determine the image plane's real dimension.

The framework's implemented procedure that computes the real size of the image to be mapped, uses the arithmetic formulas from equation 5.2.

$$\begin{aligned} Image_{width} &= 2 \times \tan(FOV/2) \times Camera\_Altitude \\ Image_{height} &= Image_{width} \times Width2Height\_Factor \end{aligned} \tag{5.2}$$

Equation 5.2: These equations compute the images' real dimensions according to the sensor's altitude. Where FOV is measured in degrees (^{o}) and the $Width2Height\_Factor$ is the sensor proportion factor to compute an image's height from its width (particularly, for a 4:3 image: $Width2Height\_Factor = \frac{3}{4}$).

For this purpose, the panel's capture altitude is also considered to compensate the distance between the above sea level reference and the ground. Since the GPS sensor measures the altitudes from above sea level ($UAVFlight\_GPSAltitude$) and the imagery dimensions are correlated with the distance from the ground to the multispectral camera.

Then, in mapping, the altitude from the initial panel's frame should be close to 0, in spite of the calibration panel's distance to the above sea level reference (*Panel_GPSAltitude*).

The field frames' altitude (*Camera_Altitude*) need to be compensated according to the above sea level panel's distance. Then, the *Camera_Altitude* in equation 5.2 is:

$$Camera\_Altitude = UAVFlight\_GPSAltitude - Panel\_GPSAltitude \tag{5.3}$$

Equation 5.3: Altitude sensor's compensation to the above sea level reference.

5.6.3.2 Coordinate and Rotation Systems

This ROS framework uses the *OccupancyGrid* message type: occupancy grid maps are meant to classify each georeferenced grid cell with an occupancy value of a certain type of terrain. For the proposed case, the values are ranged between 0 and 1, and by default, each layer is totally set to 0.5 at the beginning, representing the unknown terrain. As the procedures take place, each cell is filled with a 0 (positive) or 1 (negative).

For instance, in a vegetation layer, each vineyard row is defined by 0's, white cells that represent an occupied status; whereas the rest of the area is registered with 1's, black cells as non-vegetation.

Nevertheless, before the mapping procedures, it is important to be aware of each interface's rotation systems:

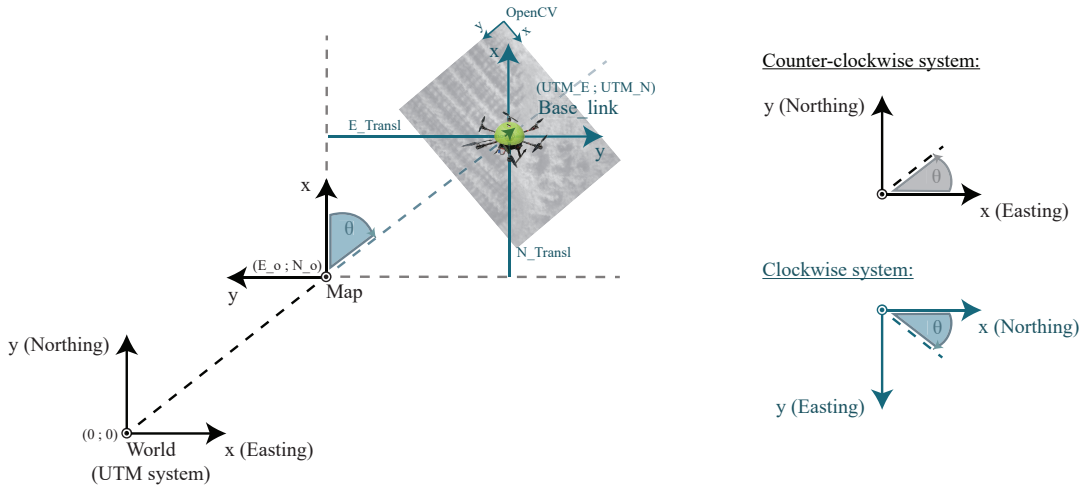


Figure 5.12: Interfaces' coordinate systems: *world* and *map* (black counter-clockwise coordinate systems), *base_link* and OpenCV (blue clockwise coordinate systems).

As illustrated in figure 5.12, the ROS interface measures according to a counter-clockwise system, with 0 facing forward along the frame's x-axis (easting). On the other hand, both OpenCV and the robot's IMU used sensor, follow the clockwise rotation system, with 0 facing forward along the frame's x-axis (northing).

The proposed model is composed by three coordinate frames: *world*, *map* and *base_link*; and they are correlated by the appropriate transformation.

The grid map frame (*map*) is directly related to the world frame(*world*) by a transformation (with 90^o counter-clockwise rotation and) translated to the UTM central position of the first captured image (E_o, N_o). Beyond that, the robot frame (*base_link*) is transformed according to the robot orientation (i.e. yaw) and translated in respect to the distance between the current UAV position and the first image location, marked by the Map origin. The implementation considered ROS counter-clockwise reference rotation system.

According to the previous coordinate system diagram 5.12, the 90^o axes rotation between the world and the map needs to be computed, equation 5.4.

$$\begin{aligned} StandardRotation : M(\theta) &= \begin{bmatrix} cos(\theta) & -sen(\theta) \\ sen(\theta) & cos(\theta) \end{bmatrix} \\ 90^o Rotated : M(\theta) &= \begin{bmatrix} cos(\theta) & -sen(\theta) \\ sen(\theta) & cos(\theta) \end{bmatrix} \cdot \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} -sen(\theta) & -cos(\theta) \\ cos(\theta) & -sen(\theta) \end{bmatrix} \end{aligned} \tag{5.4}$$

Equation 5.4: Rotation axes matrices: standard counter-clockwise and 90^o (world $\rightarrow$ map).

A pixel rotation motion according to a rotation matrix M by θ is defined in equation 5.5.

$$\begin{bmatrix} px' \\ py' \end{bmatrix} = M(\theta) \cdot \begin{bmatrix} px \\ py \end{bmatrix} \tag{5.5}$$

Equation 5.5: Point (px,py) after angle rotation(θ) to the new position (px',py')

Furthermore, as illustrated in the coordinate systems' convention diagram, after considering the 90^o clockwise rotation between the world and the map, it is assumed that the θ angle (equation 5.6) only refers to the clockwise systems transformations:

- OpenCV 2D computation $\rightarrow \pi rotation(rad)$;
- UAV Sensor Orientation $\rightarrow$ Yaw angle rotation (rad).

$$\theta = -\pi - gps\_coordinates \tag{5.6}$$

Equation 5.6: Clockwise θ angle rotation, where gps_coordinates is the Base-link IMU rotation.

Although the first captured image is centred at (E_o, N_o) position, it is required a pivot rotation around the Base_link location to accurately map the subsequent images. Each image pixel point (P(px,py)) needs to be computed according to the following steps:

1. P(px,py) point translation to map origin (E_o, N_o);
2. P(px,py) point rotation around the map origin (E_o, N_o);
3. P(px,py) point back translation to the Base_link origin (UTM_E, UTM_N);

Hence, being:

$$T(E_T; N_T) = \begin{bmatrix} 1 & 0 & E_T \\ 0 & 1 & N_T \\ 0 & 0 & 1 \end{bmatrix} \quad M(\theta) = \begin{bmatrix} -sen(\theta) & -cos(\theta) & 0 \\ cos(\theta) & -sen(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad P(px; py) = \begin{bmatrix} px \\ py \\ 1 \end{bmatrix}$$

$$P'(px'; py') = T(E_T; N_T) \cdot M(\theta) \cdot T(-E_T; -N_T) \cdot P(px; py) \tag{5.7}$$

Equation 5.7: P'(px';py') is the result of P(px;py) point pivot rotation where M is the counter-clock system's 90^o rotated matrix (equation 5.4) and T is a translation matrix between the map and the Base_link origins (E_T and N_T are respectively the *E_Transl* and *N_Transl* from figure 5.12).

From the previous pivot rotation formula it is possible to find the main layer mapping implemented algorithm to compute an accurate image position in the map:

$$\begin{bmatrix} px' \\ py' \end{bmatrix} = \begin{bmatrix} -(px - E_T) \cdot sen(\theta) - (py - N_T) \cdot cos(\theta) + E_T \\ (px - E_T) \cdot cos(\theta) - (py - N_T) \cdot sen(\theta) + N_T \end{bmatrix} \tag{5.8}$$

Equation 5.8: ROS layer mapping rotation algorithm. Accordingly, the θ angle is defined due to the clockwise required transformations, equation 5.6.

In essence, this approach is very important, not only to accurately make a stitched images map, but also to enhance the implemented dynamic map algorithm.

CHAPTER 6

Experimental Results

6.1 Experimental Setup

The Programa de Desenvolvimento Rural 2014-2020 (PDR2020) is a program co-financed by the European Structural Funds (FEADER) and comprises initiatives for rural development, pursuing the objective of promoting the competitiveness of the agro-forestry sector and the rural territories in a sustainable way. In this program's innovation and knowledge areas, the NOVA university through multiple FCT departments and the RICS group from the electrical and computer engineering department proposed several projects within the agricultural field. This research insertion in the scientific community will be very important to the following experimental results validation.

The proposed model was mainly implemented in the C++ programming language, but also supported by some implemented Python scripts. It was made fully compliant with the Robot Operating System (ROS). Tests were made with an HP EliteBook Folio 9470m dual core with 4 logical processors Intel(R) Core(TM) i5-3427U CPU 1.80GHz with 8 Gb of RAM. It is running a 64-bit Linux distribution Ubuntu 16.04.4 LTS (ROS version: Lunar), and using OpenCV 3.2.0 for low-level computer vision routines.

In order to measure the proposed model's performance, an extensive dataset of 8 missions, encompassing a total of 2475 analysed frames with a 1280×960 px resolution, have been obtained with a RedEdge-M camera carried at an approximate height between 40 m to 100 m. The dataset includes distinct terrain types such as water, foliage, rocky and sandy soils.

Although this ROS framework was developed to receive images for both offline and online missions, the following obtained experimental results only come from offline acquired datasets.

6.2 Model Parametrization

For each of the spectral indices' frames, it is necessary to parametrize each type of terrain according to the index values in those areas.

Through the resulting images, it was possible to analyse and distinguish the different field areas, allowing to parametrize the minimum and maximum values for each terrain classification. The following 6.1 tables discriminate these parameters.

Terrain Types Range Parametrization	Vegetation	Water
NDVI [MIN ; MAX]	[185 ; 255]	[123 ; 156]
ENDVI [MIN ; MAX]	[170 ; 255]	[100 ; 156]
RDVI [MIN ; MAX]	[175 ; 255]	[123 ; 140]
SR [MIN ; MAX]	[180 ; 255]	[115 ; 160]
MSAVI [MIN ; MAX]	[180 ; 255]	[122 ; 134]

Terrain Types Range Parametrization	Soil	Rocks
NDVI [MIN ; MAX]	[70 ; 140]	[50 ; 105]
ENDVI [MIN ; MAX]	[95 ; 147]	[50 ; 120]
RDVI [MIN ; MAX]	[85 ; 140]	[40 ; 108]
SR [MIN ; MAX]	[35 ; 125]	[50 ; 105]
MSAVI [MIN ; MAX]	[35 ; 125]	[40 ; 100]

Table 6.1: Terrain classification parametrization.

6.3 Experimental Datasets

The infrastructure involved will be closely tested in the field (figure 6.1), in real-time, as a central object of study responsible to acquire every kind of data involved with the mission in progress.

Many agricultural producers and organizations involved with the current project supported this work with their own productions as samples for testing. For instance, there were collected images from vineyards in Palmela, Roxo production, orchards, potatoes and tomatoes productions and also from the FRUTALVOR central which is an enriched organization with their own orchards and horticultural productions, nationally recognized and distributed throughout the country.

As a result, the Rededge-M sensor attached to the UAV, will capture several datasets from the crops in study. Furthermore, both for real-time and offline missions, the datasets

a b c

Figure 6.1: Experimental testing in the field: (a) UAV infrastructure, (b)Vineyard in Palmela and (c) Micasense calibration panel.

will be stored at the SD Card included in the camera kit. Every time the camera is powered up, a new folder is created with another folder inside named "000", [31]. The latter mentioned folder, supports until 999 files within, where each capture is composed by 5 different image files, each one with a suffix from 1 to 5, corresponding to the Blue, Green, Red, NIR and Rededge bands, following the respective order. See table 4.1 for more details.

Accordingly, the illustrations below present the imagery nomenclature of the Micasense sensor in use:

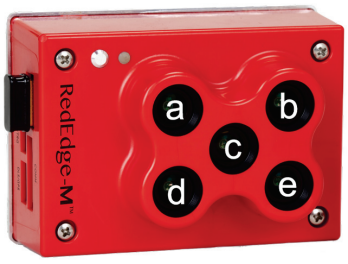


Figure 6.2: Imagery nomenclature of the Rededge-M sensor by Micasense

Furthermore, this imagery nomenclature is followed by an example of 5 files belonging to a single capture from the same camera. Each image is in agreement with the corresponding lenses' letters (Figure 6.2). The respective band filters are specified as follows in illustration 6.3.

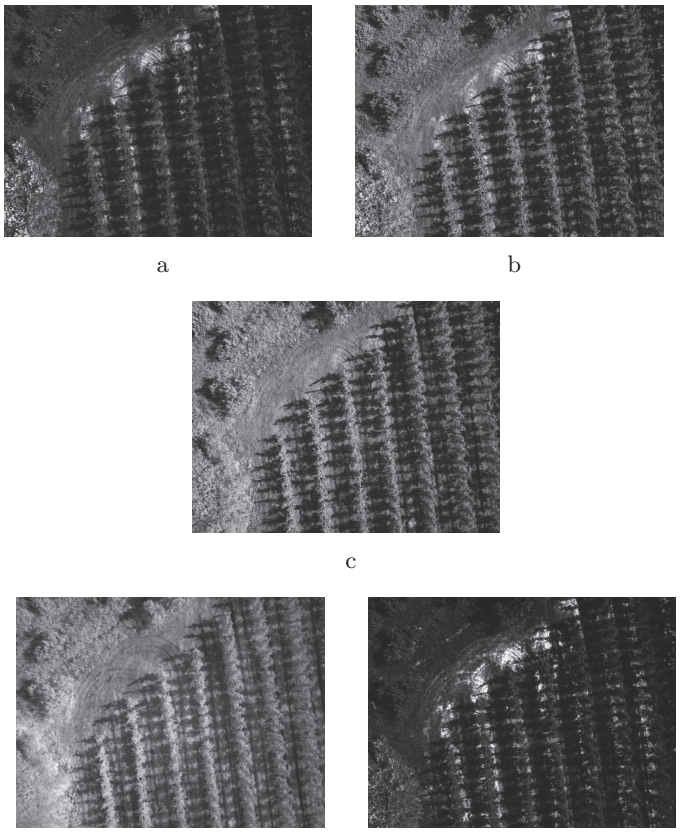


Figure 6.3: Relative provision between the band filters: (a) Blue, (b) Green, (c) RE, (d) NIR and (e) Red.

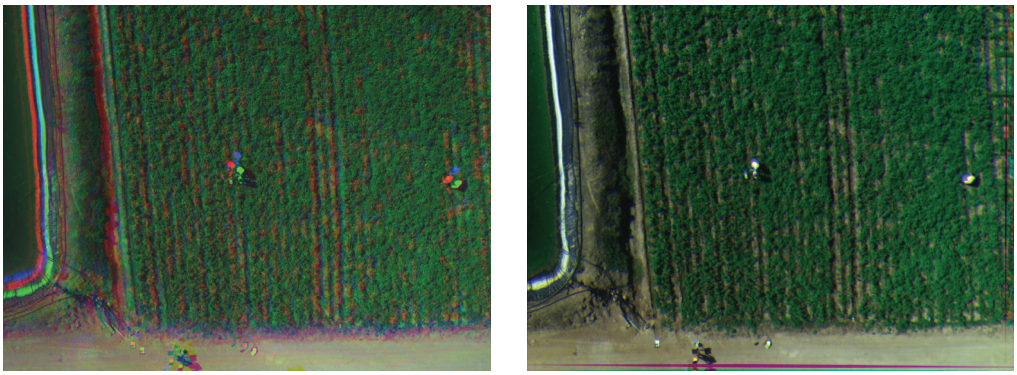
The capture illustrated in 6.3 belongs to a dataset from one of the FRUTALVOR vineyards. Although it was shot from a 82.2 m height above the sea level, it is still possible to verify an apprehensible misalignment between the band filters due to an hardware conception of the Micasense sensor, since the current lenses reserve a certain distance from each other (as explained in 4.2.5).

6.4 Results Validation

6.4.1 Colour Band Alignments

Before any further terrain classification, as previously mentioned in this document, it is necessary to align the images captured by the different lenses of the multispectral camera.

Figure 6.4 illustrates both best and worst case scenarios related to the alignments. The former reflects an inaccurate alignment not allowing the image to be correctly understood and later processed, whereas the latter represents a fully successful RGB channels alignment, contributing to the next stages of this framework.



a b

Figure 6.4: Alignments examples.

6.4.2 Terrain Classification Assessment

For the scope of this project, several datasets were studied. However, only eight of these were indeed tested (annex B).

For each dataset, the five implemented spectral indices (NDVI, ENDVI, RDVI, MSAVI and SR) were tested; each of these spectral indices spread into the four analysed terrains (i.e. vegetation, water, rocks and soil). This processing allowed to calculate and obtain a successful rate based on the following classified characteristics:

- True Positives (True POS): The algorithm classified the collected information as a certain terrain → the observation of this terrain validated the result;
- True Negatives (True NEG): The algorithm classified the collected information as not being a certain terrain → the observation of this terrain validated the result;
- False Positives (False POS): The algorithm classified the collected information as a certain terrain → the observation of this terrain did not validate the result;
- False Negatives (False NEG): The algorithm classified the collected information as not being a certain terrain → the observation of this terrain did not validate the result.

All the mentioned characteristics were weighted according to the respective relevance in each analysed dataset. The successful rate was calculated as follows:

$$Success = TruePOS \times \frac{RedPixels}{AllPixels} + TrueNEG \times \frac{OtherPixels}{AllPixels}(\%) \tag{6.1}$$

Equation 6.1: Successful rate formula taking into account the image's presented terrain weight.

As figure 6.5 shows, every terrain was identified and coloured in order to represent the validation masks (Lourinhã potatoes crop presented as follows).

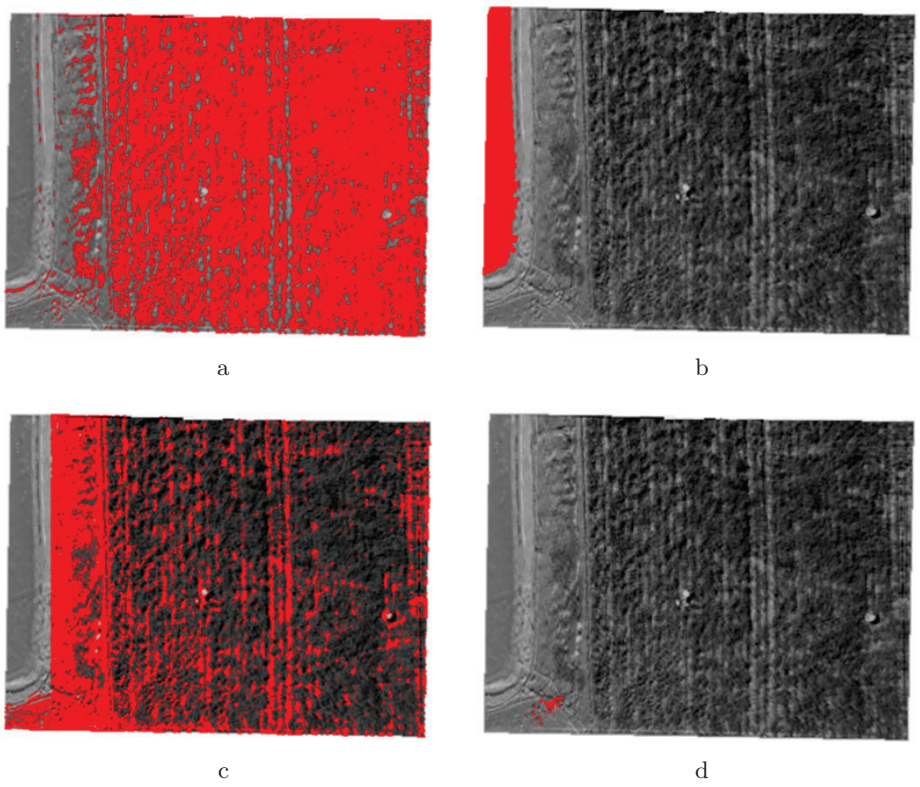


Figure 6.5: Validation masks for the implemented ROS framework terrain classification results.

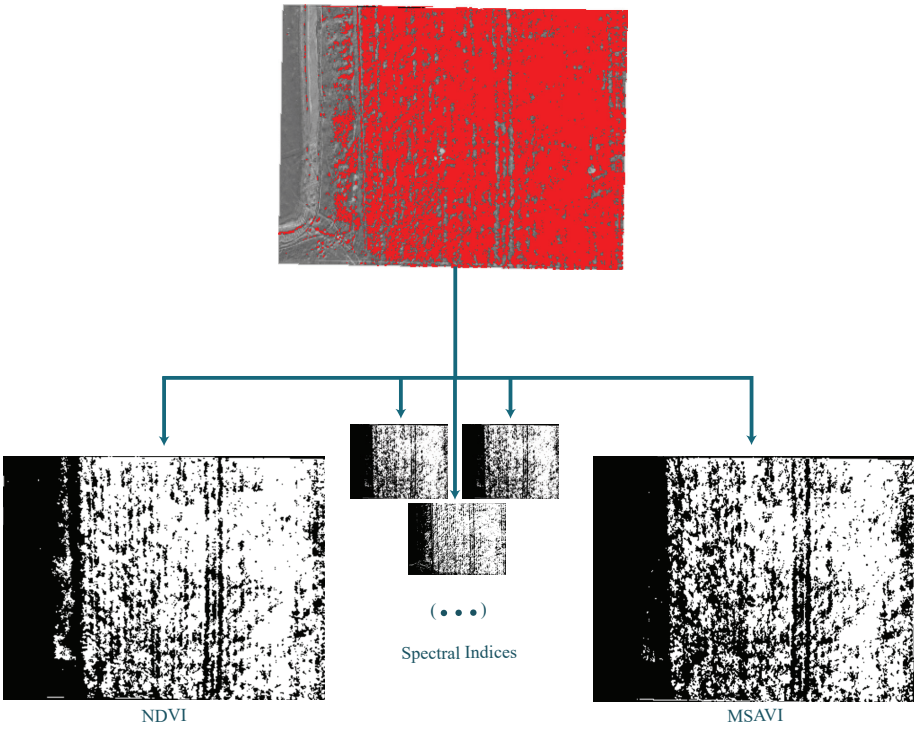


Figure 6.6: Spectral Indices Results for vegetation.

This field clearly distinguishes the different terrains into the four types described, allowing the drone to observe and collect the information for users. In this image, rocks weight much less than vegetation: the coloured representation of rocks versus vegetation is, approximately, 2% and 90% respectively (figure 6.5).

In order to achieve the global success rate (based on annex B test detailed results), the average of each terrain successful outcome was calculated according to each spectral index result (figure 6.6):

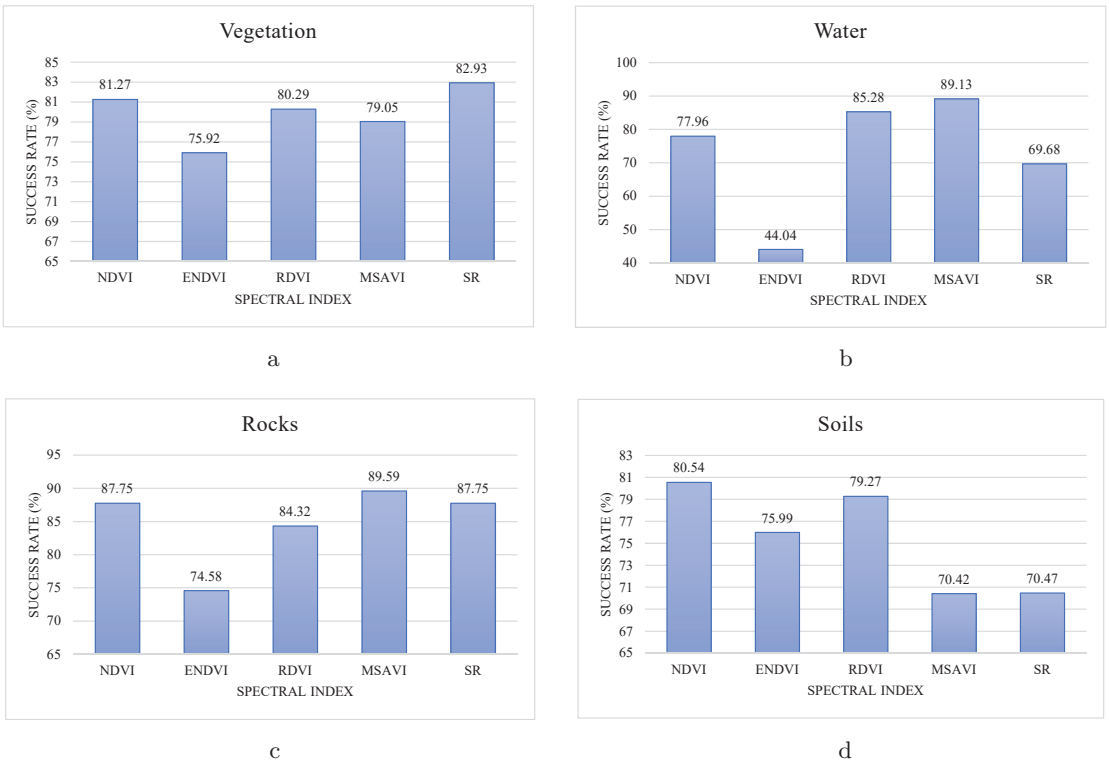


Figure 6.7: Terrain classification success rate results: Vegetation (a), Water (b), Rocks (c) and Soils (d).

In the end, the achieved results (figure 6.7) reflected a good margin to work on, since the majority of the obtained percentages outgrew an average of 80% of success. Another possible outcome from these graphs would be the best spectral index to apply to a particular terrain, despite all spectral indices apply to each terrain. However, according to the results, vegetation, water, rocks and soils would be best recognized, respectively, by SR, MSAVI, MSAVI and NDVI indices.

CHAPTER 7

Conclusions and Future Work

7.1 Conclusions

According to the purpose of this project, analysing and getting the information from agricultural fields provide the possibility to enhance farmers experience, as well autonomous vehicle navigation and multi-robot cooperation, facing nowadays changing technological world. The ability to recognize the terrains enables to take full advantage of technology, introducing ancient topography techniques to new autonomous, fast, accurate, precise and unmanned systems. It is crucial to keep up with autonomous robots, which are embracing society, aiding obstacles' detection, safe landing and terrain adaptation. Internet of Things (IOT) allows robot's networking and information sharing, enabling the cooperation between several machines, for instance, to apply different drones' altitudes and increase the observed area range.

This project is focused on computer vision analysis that collects the most interesting characteristics acquired from a multispectral camera and transforms this data into valuable and readable information for users. The obtained images require full alignment, due to the used lenses different positions. The RedEdge-M raw imagery goes through a certain correction processing in order to interpret the reflected flux of solar radiation captured by the camera. Afterwards, the analysed band frames are combined into several spectral indices capable of identifying terrain singularities, such as vegetation, water, rocks and soil. These parameters accuracy provide the correspondent results clarification and validation, through training masks.

After the terrain classification, the information will be spread into semantic layered maps. Each layer has a unique classification, applying to land features distinction or even agriculture precision ends, with crops growing and health status.

Although the global overview reflected a good performance and output, the multiple used methods raised some challenges, such as reaching the perfect lenses' alignment, obtaining the real measurements to build dynamic semantic maps and sharing an interoperable system between different infrastructures and technologies that demanded complex coordinates' transformation algorithms.

The developed open-source ROS framework allowed to obtain results, validated together with PDR2020 projects, from which was possible to achieve successful output data according to terrain types classification. The efficiency was tested for each implemented spectral index: reaching an average of, approximately, 80% of successful rates.

7.2 Future Work

The purpose of this project covers a wide range of use cases, shaping the most diverse application scenarios. Although this approach is meant for different terrains' classification, there is still plenty issues to be concerned with, processing improvements and future assets to address in this research field. The following topics introduce some of these enhancements:

- Improve the library algorithms' performance handled by Micasense lenses corrections and distortions;
- Use the applied horizontal plane recognition, provided by yaw, as well as the vertical plane recognition expansion, considering roll and pitch orientations, allowing a 3D terrain modulation;
- Odroid integration in UAV infrastructure in order to provide faster processing results mainly during online missions, since ethernet connection is made possible;
- Spectral indices deep study, focused on nutritional analysis, crops' growth and development status (e.g. estimating crops' harvest periods) and plagues or diseases detection (e.g. improve countermeasure proactivity);
- Spread multispectral analysis and classification to livestock activity usage.

Many other features can be improved and implemented, since this computer vision processing and mapping field stands for an important attending issue in the future and aspires to a growing impact in society, from health to agricultural production quality and terrain nutrition, fertilization and management. The enhancement of different land classification techniques, together with fast and accurate systems, intend to provide agile and autonomous methods to implement and expand Precision Agriculture.

Bibliography

[1] J. Beddington. "Food security: contributions from science to a new and greener revolution". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 365.1537 (2010), pp. 61–71. ISSN: 0962-8436. DOI: 10.1098/rstb.2009.0201. URL: http://rstb.royalsocietypublishing.org/cgi/doi/10.1098/rstb.2009.0201.

[2] J. Foley, G. Steinmetz, and J. Richardson. "Cinco propostas para Alimentar o mundo — O Futuro da Alimentação". In: *National Geographic Portugal* 158 (2014), pp. 1–31.

[3] X. E. Yang, W. R. Chen, and Y. Feng. "Improving human micronutrient nutrition through biofortification in the soil-plant system: China as a case study". In: *Environmental Geochemistry and Health* 29.5 (2007), pp. 413–428. ISSN: 02694042. DOI: 10.1007/s10653-007-9086-0.

[4] S. Thrikawala, A. Weersink, G. Kachanoski, and G. Fox. "ECONOMIC FEASIBILITY OF VARIABLE-RATE TECHNOLOGY FOR NITROGEN ON CORN". In: 81.November (1999), pp. 914–927.

[5] C. Zhang and J. M. Kovacs. "The application of small unmanned aerial systems for precision agriculture: A review". In: *Precision Agriculture* 13.6 (2012), pp. 693–712. ISSN: 13852256. DOI: 10.1007/s11119-012-9274-5.

[6] MOFAR. "MO dular and Collaborating Autonomous FAR mer Vehicles for the Agriculture of the Future Table of Contents". In: *Mofar* 1.1 (), pp. 1–69.

[7] C. C. Lelong, P. Burger, G. Jubelin, B. Roux, S. Labbé, and F. Baret. "Assessment of unmanned aerial vehicles imagery for quantitative monitoring of wheat crop in small plots". In: *Sensors* 8.5 (2008), pp. 3557–3585. ISSN: 14248220. DOI: 10.3390/s8053557.

[8] A. S. Laliberte and A. Rango. "Image Processing and Classification Procedures for Analysis of Sub-decimeter Imagery Acquired with an Unmanned Aircraft over Arid Rangelands". In: *GIScience & Remote Sensing* 48.1 (2011), pp. 4–23. ISSN: 1548-1603. DOI: 10.2747/1548-1603.48.1.4. URL: http://bellwether.metapress.com/openurl.asp?genre=article{\&}id=doi:10.2747/1548-1603.48.1.4.

[9] H. Eisenbeiss. "A mini unmanned aerial vehicle (UAV): system overview and image acquisition". In: *Proceedings of the International Workshop on Processing and Visualization using High-Resolution Imagery* (2004), pp. 1–7. ISSN: 2194-9050. DOI: 10.1017/S0003598X00047980. URL: http://www.isprs.org/proceedings/XXXVI/5-W1/papers/11.pdf.

[10] S Nebiker, A Annen, M Scherrer, and D Oesch. "A light-weight multispectral sensor for micro UAV—Opportunities for very high resolution airborne remote sensing". In: *International Arch. Photogrammetry and Remote Sensing Spatial Information* 37 (2008), pp. 1193–1198.

[11] D. A. Stow. "Remote Sensing". In: *Research Methods in Geography* 14.1 (2010), pp. 155–172. DOI: 10.3390/rs4061856. URL: http://www.wiley.com/WileyCDA/WileyTitle/productCd-1405107111.html.

[12] A. Mora, T. Santos, S. Łukasik, J. Silva, A. Falcão, J. Fonseca, and R. Ribeiro. "Land Cover Classification from Multispectral Data Using Computational Intelligence Tools: A Comparative Study". In: *Information* 8.4 (2017), p. 147. ISSN: 2078-2489. DOI: 10.3390/info8040147. URL: http://www.mdpi.com/2078-2489/8/4/147.

[13] *NASA's Landsat 5 Satellite Sets Guinness World Record*. URL: https://www.space.com/19757-landsat-5-satellite-nasa-world-record.html (visited on 02/25/2018).

[14] *LANDSAT 7 ETM Satellite Sensor | Satellite Imaging Corp.* URL: https://www.satimagingcorp.com/satellite-sensors/other-satellite-sensors/landsat/ (visited on 02/25/2018).

[15] E. Adam, O. Mutanga, and D. Rugege. "Multispectral and hyperspectral remote sensing for identification and mapping of wetland vegetation: A review". In: *Wetlands Ecology and Management* 18.3 (2010), pp. 281–296. ISSN: 09234861. DOI: 10.1007/s11273-009-9169-z.

[16] *Multispectral vs Hyperspectral Imagery Explained - GIS Geography*. URL: http://gisgeography.com/multispectral-vs-hyperspectral-imagery-explained/ (visited on 02/28/2018).

[17] *AVIRIS - Airborne Visible / Infrared Imaging Spectrometer - Imaging Spectroscopy*. URL: https://aviris.jpl.nasa.gov/aviris/imaging_spectroscopy.html (visited on 02/28/2018).

[18] J. A. J. Berni, P. J. Zarco-Tejada, L. Suárez, and E. Fereres. "Thermal and Narrowband Multispectral Remote Sensing for Vegetation Monitoring From an Unmanned Aerial Vehicle". In: *IEEE Transactions on Geoscience and Remote Sensing* 47.3 (2009), pp. 722–738. ISSN: 0196-2892. DOI: 10.1109/TGRS.2008.2010457.

[19] P. J. Zarco-Tejada, J. A. Berni, L. Suárez, G. Sepulcre-Cantó, F. Morales, and J. R. Miller. "Imaging chlorophyll fluorescence with an airborne narrow-band multispectral camera for vegetation stress detection". In: *Remote Sensing of Environment* 113.6 (2009), pp. 1262–1275. ISSN: 00344257. DOI: 10.1016/j.rse.2009.02.016. URL: http://dx.doi.org/10.1016/j.rse.2009.02.016.

[20] *Tetracam - Recommendations for Band Pass Filter Selections*. URL: http://www.tetracam.com/Tetracam-RecommendationsforBandPassFilterSelection.htm (visited on 02/28/2018).

[21] *What is NDVI » Hummingbird Aerial Surveys*. URL: http://hummingbirdaerialsurveys.com/about/what-is-ndvi/ (visited on 02/28/2018).

[22] D. Haboudane, J. R. Miller, E. Pattey, P. J. Zarco-Tejada, and I. B. Strachan. "Hyperspectral vegetation indices and novel algorithms for predicting green LAI of crop canopies: Modeling and validation in the context of precision agriculture". In: *Remote Sensing of Environment* 90.3 (2004), pp. 337–352. ISSN: 00344257. DOI: 10.1016/j.rse.2003.12.013.

[23] M. D. Pritt. "Fast orthorectified mosaics of thousands of aerial photographs from small UAVs". In: *Proceedings - Applied Imagery Pattern Recognition Workshop* 2015-February.February (2015). ISSN: 21642516. DOI: 10.1109/AIPR.2014.7041928.

[24] D. Carneiro, D. Carneiro, D. A. Silva, G. Willem, A. Toonstra, H. Lacet, and S. Souza. "De Vant Processados Com Os Softwares". In: April (2017).

[25] *How to verify that there is Enough Overlap between the Images – Support*. URL: https://support.pix4d.com/hc/en-us/articles/203756125-How-to-verify-that-there-is-Enough-Overlap-between-the-Images (visited on 03/02/2018).

[26] *Pix4Dmapper: Photogrammetry software. Desktop or cloud processing*. URL: https://pix4d.com/product/pix4dmapper-photogrammetry-software/ (visited on 03/02/2018).

[27] *Big-O Algorithm Complexity Cheat Sheet (Know Thy Complexities!) @ericdrowell*. URL: http://bigocheatsheet.com/ (visited on 03/03/2018).

[28] M. Quigley, B. Gerkeyy, K. Conleyy, J. Fausty, T. Footey, J. Leibsz, E. Bergery, R. Wheelery, and A. Ng. "ROS: an open-source Robot Operating System". In: *Computer Science Department, Stanford University, Stanford, CA* Figure 1 (2009). ISSN: 0165-022X. DOI: http://www.willowgarage.com/papers/ros-open-source-robot-operating-system. arXiv: 1106.4561.

[29] I. Culjak, D. Abram, T. Pribanic, H. Dzapo, and M. Cifrek. "A brief introduction to OpenCV". In: *MIPRO, 2012 Proceedings of the 35th International Convention* (2012), pp. 1725–1730. DOI: 978-1-4673-2577-6. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6240859.

[30] U. Manual. *Downwelling Light Sensor (DLS) Integration Guide and User Manual*. 2016.

[31] "MicaSense RedEdge ™ 3 Multispectral Camera User Manual". In: October (2015), pp. 1–33.

[32] *C++ Interface for ExifTool (by Phil Harvey)*. URL: http://owl.phy.queensu.ca/~phil/cpp_exiftool/ (visited on 09/01/2018).

[33] *RedEdge Camera Radiometric Calibration Model – MicaSense Knowledge Base*. URL: https://support.micasense.com/hc/en-us/articles/115000351194-Rededge-Camera-Radiometric-Calibration-Model#model (visited on 08/30/2018).

[34] *MicaSense Image Processing Tutorial 1*. URL: https://micasense.github.io/imageprocessing/MicaSenseImageProcessingTutorial1.html (visited on 08/30/2018).

[35] "ESA - Eduspace EN - Global Change - Optical properties of ice and snow". In: (2013). URL: https://www.esa.int/SPECIALS/Eduspace_Global_EN/SEMPJ7TWLUG_0.html#subhead2.

[36] *Image Alignment (ECC) in OpenCV (C++ / Python) | Learn OpenCV*. URL: https://www.learnopencv.com/image-alignment-ecc-in-opencv-c-python/ (visited on 09/03/2018).

[37] *Features2D + Homography to find a known object — OpenCV 2.4.13.7 documentation*. URL: https://docs.opencv.org/2.4/doc/tutorials/features2d/feature_homography/feature_homography.html (visited on 09/07/2018).

[38] H. Bay, T. Tuytelaars, and L. Van Gool. "SURF: Speeded up robust features". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 3951 LNCS (2006), pp. 404–417. ISSN: 03029743. DOI: 10.1007/11744023_32.

[39] *Feature Matching with FLANN — OpenCV 2.4.13.7 documentation*. URL: https://docs.opencv.org/2.4/doc/tutorials/features2d/feature_flann_matcher/feature_flann_matcher.html (visited on 09/10/2018).

[40] *grid_map - ROS documentation*. URL: https://github.com/anybotics/grid_map (visited on 09/13/2018).

[41] *sensor_msgs/Image Documentation*. URL: http://docs.ros.org/api/sensor_msgs/html/msg/Image.html (visited on 09/15/2018).

[42] *sensor_msgs/NavSatFix Documentation*. URL: http://docs.ros.org/api/sensor_msgs/html/msg/NavSatFix.html (visited on 09/15/2018).

APPENDIX A

Rededge-M Imagery Sensor

A.1 Sample of a YAML metadata file from Rededge-M Imagery Sensor

ExifToolVersion: "10.10"

FileName: "IMG_0004_4.tif"

Directory: /home/beatrizsalvado/Desktop/Mission_1_DataSet/28_05_2018/0016SET/000

FileSize: "2.4 MB"

FilePermissions: rw-rw-r–

FileModifyDate: "2018:05:28 03:40:40+01:00"

FileAccessDate: "2018:07:03 11:13:11+01:00"

FileInodeChangeDate: "2018:05:28 11:49:50+01:00"

FileType: TIFF

FileTypeExtension: tif

MIMEType: image/tiff

ExifByteOrder: "Little-endian (Intel, II)"

SubfileType: Full-resolution Image

ImageWidth: "1280"

ImageHeight: "960"

BitsPerSample: "16"

Compression: Uncompressed

PhotometricInterpretation: BlackIsZero

StripOffsets: "(Binary data 70 bytes, use -b option to extract)"

Orientation: Horizontal (normal)

SamplesPerPixel: "1"

RowsPerStrip: "100"

StripByteCounts: "(Binary data 69 bytes, use -b option to extract)"

PlanarConfiguration: Chunky

Software: "v3.3.0"

BlackLevelRepeatDim: "2 2"

BlackLevel: "4800 4800 4800 4800"

OpcodeList3: "(Binary data 184 bytes, use -b option to extract)"

Make: MicaSense

Model: RedEdge-M

ModifyDate: "2018:05:28 09:54:20"

XMPToolkit: "XMP Core 4.4.0"

About: Pix4D Camera Information

BootTimestamp: "1050"

RadiometricCalibration: "0.00018989564643105694, 8.4584105753599054e-08, 1.5627073908987655e-05"

FlightId: yNRal1Q9wdZ52FgTCbA6

CaptureId: "474EcftDp49zLLSFcVvf"

TriggerMethod: "1"

PressureAlt: "57.867588043212891"

DarkRowValue: "5059, 5083, 5056, 5040"

BandName: NIR

CentralWavelength: "840"

WavelengthFWHM: "40"

VignettingCenter: "617.11652352677311, 474.27859714857334"

VignettingPolynomial: "-0.00011825707746914148, 5.3870492875556434e-08, -1.2798756344064433e-08, 4.5220705895394507e-11, -5.9411243402537197e-14, 2.7130384554409001e-17"

ModelType: perspective

PrincipalPoint: "2.37564,1.71116"

PerspectiveFocalLength: "5.4537928457993647"

PerspectiveFocalLengthUnits: mm

PerspectiveDistortion: "-0.10451232003408055, 0.14745719362277598, -0.0098209126277837705, 0.00021427853748014447, 0.00018911288706109603"

BandSensitivity: "0.20052625875524618"

RigCameraIndex: "3"

IrradianceExposureTime: "0.10100000351667404"

IrradianceGain: "16"

Irradiance: "0.57690227031707764"

IrradianceYaw: "159.65152843760467"

IrradiancePitch: "1.0929077884431513"

IrradianceRoll: "-0.94607740196678425"

-Yaw: "175.40192165970376"

-Pitch: "1.6082171320641621"

-Roll: "-0.86810295624345779"

Serial: DL05-1732217-SC

SwVersion: "v1.0.1"

SensorId: "3"

CenterWavelength: "840"

Bandwidth: "40"

TimeStamp: "113289"

Exposure: "0.10100000351667404"

Gain: "16"

SpectralIrradiance: "0.57690227031707764"

RawMeasurement: "3079"

OffMeasurement: "4718"

Yaw: "2.7864448270775597"

Pitch: "0.019074839329022621"

Roll: "-0.016512165643034264"

ExposureTime: "1/1034"

FNumber: "2.8"

ISOSpeed: "100"

ExifVersion: "0230"

SerialNumber: RM01-1743095-SC

ExposureProgram: Program AE

MeteringMode: Multi-spot

FocalLength: "5.5 mm"

FocalPlaneXResolution: "266.666667"

FocalPlaneYResolution: "266.666667"

FocalPlaneResolutionUnit: mm

DateTimeOriginal: "2018:05:28 09:54:20"

CreateDate: "2018:05:28 09:54:20"

SubSecTime: "171000000"

GPSVersionID: "2.2.0.0"

GPSLatitudeRef: North

GPSLongitudeRef: West

GPSAltitudeRef: Above Sea Level

GPSDOP: "0"

GPSLatitude: "38 deg 39’ 32.22""

GPSLongitude: "9 deg 12’ 20.57""

GPSAltitude: "93.91 m"

Aperture: "2.8"

ImageSize: "1280x960"

Megapixels: "1.2"

ShutterSpeed: "1/1034"

FOV: "48.8 deg"

-GPSAltitude: "93.9 m Above Sea Level"

-GPSLatitude: "38 deg 39’ 32.22" N"

-GPSLongitude: "9 deg 12’ 20.57" W"

GPSPosition: "38 deg 39’ 32.22" N, 9 deg 12’ 20.57" W"

ScaleFactor35efl: "7.2"

CircleOfConfusion: "0.004 mm"

FocalLength35efl: "5.5 mm (35 mm equivalent: 39.7 mm)"

HyperfocalDistance: "2.59 m"

SubSecModifyDate: "2018:05:28 09:54:20.171000000"

APPENDIX B

Terrain Classification Results

This section presents the overall results, discriminated in the following tables. The obtained values describe the terrain classification output according to each type and spectral index. The table below (table B.1) resumes all the average outcome resulting from its following tables.

	MEAN SUCCESS (%)
NDVI	
Vegetation	81.27
Rocks	87.75
Water	77.96
Soils	80.54
ENDVI	
Vegetation	75.92
Rocks	74.58
Water	44.04
Soils	75.99
RDVI	
Vegetation	80.29
Rocks	84.32
Water	85.28
Soils	79.27
MSAVI	
Vegetation	79.05
Rocks	89.59
Water	89.13
Soils	70.42
SR	
Vegetation	82.93
Rocks	87.75
Water	69.68
Soils	70.47

Table B.1: Overall average results

	DATASET									
	Mission 63					Lourinhã Potatoes				
	True POS	True NEG	False POS	False NEG	% SUCCESS	True POS	True NEG	False POS	False NEG	% SUCCESS
NDVI										
Vegetation	69.54	99.43	30.46	0.57	94.45	78.17	76.90	21.83	23.10	77.70
Rocks	-	99.38	-	0.62	99.38	0.93	98.40	99.07	1.60	98.29
Water	-	45.02	-	54.98	45.02	52.86	85.16	47.14	14.84	83.47
Soils	56.99	97.95	43.01	2.05	65.64	34.92	87.87	65.08	12.13	74.15
ENDVI										
Vegetation	62.33	92.32	37.67	7.68	87.32	93.61	100.00	6.39	0.00	96.01
Rocks	-	88.03	-	11.97	88.03	65.42	95.29	34.58	4.71	95.26
Water	-	36.80	-	63.20	36.80	98.79	76.70	1.21	23.30	77.86
Soils	58.78	80.61	41.22	19.39	63.39	49.89	89.68	50.11	10.32	79.37
RDVI										
Vegetation	38.14	99.42	61.86	0.58	89.21	73.81	83.66	26.19	16.34	77.51
Rocks	-	98.37	-	1.63	98.37	0.00	98.49	100.00	1.51	98.38
Water	-	65.35	-	34.65	65.35	68.81	88.82	31.19	11.18	87.77
Soils	61.37	96.63	38.63	3.37	68.81	39.65	83.88	60.35	16.12	72.43
MSAVI										
Vegetation	29.85	99.48	70.15	0.52	87.87	69.57	86.21	30.43	13.79	75.81
Rocks	-	98.83	-	1.17	98.83	0.00	99.42	100.00	0.58	99.32
Water	-	76.83	-	23.18	76.83	45.62	91.08	54.38	8.92	88.69
Soils	20.24	92.37	79.76	7.63	35.47	17.24	94.32	82.76	5.68	74.35
SR										
Vegetation	83.21	99.38	16.79	0.62	96.69	81.36	73.54	18.64	26.46	78.43
Rocks	-	99.38	-	0.62	99.38	0.93	98.40	99.07	1.60	98.29
Water	-	29.55	-	70.45	29.55	70.16	78.70	29.84	21.30	78.25
Soils	20.26	96.74	79.74	3.26	36.40	18.61	93.58	81.39	6.42	74.16

	DATASET									
	Mission 18					Palmela 02				
	True POS	True NEG	False POS	False NEG	% SUCCESS	True POS	True NEG	False POS	False NEG	% SUCCESS
NDVI										
Vegetation	78.70	85.59	21.30	14.41	81.99	36.05	95.45	63.95	4.55	71.97
Rocks	19.48	95.06	80.52	4.94	94.82	-	64.58	-	35.42	64.58
Water	-	81.83	-	18.17	81.83	-	86.99	-	13.01	86.99
Soils	61.52	96.24	38.48	3.76	83.87	82.70	69.73	17.30	30.27	78.54
ENDVI										
Vegetation	84.08	98.70	15.92	1.30	91.06	21.19	99.31	78.81	0.69	68.43
Rocks	63.64	89.00	36.36	11.00	88.92	-	48.45	-	51.55	48.45
Water	-	63.35	-	36.65	63.35	-	36.86	-	63.14	36.86
Soils	72.80	94.55	27.20	5.45	86.80	84.73	76.37	15.27	23.63	82.05
RDVI										
Vegetation	70.39	89.94	29.61	10.06	79.73	38.74	94.87	61.26	5.13	72.68
Rocks	19.48	94.53	80.52	5.47	94.29	-	59.81	-	40.19	59.81
Water	-	87.32	-	12.68	87.32	-	91.44	-	8.56	91.44
Soils	65.27	95.51	34.73	4.49	84.74	75.73	71.46	24.27	28.54	74.36
MSAVI										
Vegetation	64.18	91.47	35.82	8.53	77.22	35.53	95.43	64.47	4.57	71.76
Rocks	12.99	96.83	87.01	3.17	96.57	-	73.29	-	26.71	73.29
Water	-	91.08	-	8.92	91.08	-	93.16	-	6.84	93.16
Soils	38.61	98.20	61.39	1.80	76.98	76.60	75.93	23.40	24.07	76.38
SR										
Vegetation	83.79	82.71	16.21	17.29	83.28	39.67	94.79	60.33	5.21	73.01
Rocks	19.48	95.06	80.52	4.94	94.82	-	64.58	-	35.42	64.58
Water	-	75.03	-	24.97	75.03	-	78.95	-	21.05	78.95
Soils	39.13	98.27	60.87	1.73	77.21	76.11	75.99	23.89	24.01	76.07

	DATASET									
	Palmela 08					Palmela 10				
	True POS	True NEG	False POS	False NEG	% SUCCESS	True POS	True NEG	False POS	False NEG	% SUCCESS
NDVI										
Vegetation	87.97	96.76	12.03	3.24	94.40	60.12	99.57	39.88	0.43	81.32
Rocks	-	76.69	-	23.31	76.69	-	77.91	-	22.09	77.91
Water	-	87.18	-	12.82	87.18	-	89.37	-	10.63	89.37
Soils	81.79	99.95	18.21	0.05	87.18	86.91	99.93	13.09	0.07	92.17
ENDVI										
Vegetation	60.03	95.13	39.97	4.87	85.71	20.30	97.64	79.70	2.36	61.86
Rocks	-	66.97	-	33.03	66.97	-	56.44	-	43.56	56.44
Water	-	36.41	-	63.59	36.41	-	31.74	-	68.26	31.74
Soils	80.09	87.85	19.91	12.15	82.40	85.53	69.16	14.47	30.84	78.92
RDVI										
Vegetation	89.18	97.24	10.83	2.76	95.07	61.58	99.58	38.42	0.42	82.00
Rocks	-	67.97	-	32.03	67.97	-	67.16	-	32.84	67.16
Water	-	91.10	-	8.90	91.10	-	93.64	-	6.36	93.64
Soils	75.84	100.00	24.16	0.00	83.02	80.69	99.89	19.31	0.11	88.44
MSAVI										
Vegetation	85.43	97.19	14.57	2.82	94.03	57.05	99.59	42.95	0.41	79.91
Rocks	-	78.90	-	21.10	78.90	-	77.59	-	22.41	77.59
Water	-	92.46	-	7.54	92.46	-	94.90	-	5.10	94.90
Soils	72.75	99.48	27.25	0.52	80.69	79.22	99.33	20.78	0.67	87.34
SR										
Vegetation	91.62	96.06	8.38	3.94	94.87	66.43	99.56	33.57	0.44	84.23
Rocks	-	76.69	-	23.31	76.69	-	77.91	-	22.09	77.91
Water	-	77.22	-	22.78	77.22	-	80.96	-	19.04	80.96
Soils	71.34	99.76	28.66	0.24	79.78	79.02	100.00	20.98	0.00	87.50

	DATASET									
	Mission in Roxo field					Vinha				
	True POS	True NEG	False POS	False NEG	% SUCCESS	True POS	True NEG	False POS	False NEG	% SUCCESS
NDVI										
Vegetation	83.19	99.10	16.81	0.90	92.46	64.32	48.41	35.68	51.59	55.84
Rocks	-	95.74	-	4.26	95.74	-	94.58	-	5.42	94.58
Water	-	60.65	-	39.35	60.65	-	89.18	-	10.82	89.18
Soils	72.98	100.00	27.02	0.00	83.70	38.02	100.00	61.98	0.00	79.11
ENDVI										
Vegetation	17.06	98.02	82.94	1.98	64.21	18.62	82.62	81.38	17.38	52.72
Rocks	-	71.75	-	28.25	71.75	-	80.84	-	19.16	80.84
Water	-	23.27	-	76.73	23.27	-	46.04	-	53.96	46.04
Soils	84.95	51.59	15.05	48.41	71.73	58.60	65.58	41.40	34.42	63.23
RDVI										
Vegetation	87.03	98.46	12.97	1.54	93.69	39.92	63.47	60.08	36.53	52.47
Rocks	-	93.95	-	6.05	93.95	-	94.62	-	5.38	94.62
Water	-	73.14	-	26.86	73.14	-	92.45	-	7.55	92.45
Soils	69.78	99.74	30.22	0.26	81.66	43.11	99.83	56.89	0.17	80.71
MSAVI										
Vegetation	83.44	98.79	16.56	1.21	92.38	32.65	71.61	67.35	28.39	53.41
Rocks	-	95.99	-	4.01	95.99	-	96.25	-	3.75	96.25
Water	-	80.83	-	19.17	80.83	-	95.07	-	4.93	95.07
Soils	29.41	98.58	70.59	1.42	56.83	27.42	99.63	72.58	0.38	75.29
SR										
Vegetation	87.58	99.10	12.42	0.90	94.29	72.66	46.43	27.34	53.57	58.69
Rocks	-	95.74	-	4.26	95.74	-	94.58	-	5.42	94.58
Water	-	52.02	-	47.98	52.02	-	85.43	-	14.57	85.43
Soils	28.55	99.67	71.45	0.33	56.75	28.55	99.99	71.45	0.01	75.91