Università degli Studi di Brescia

Dipartimento di Ingegneria Meccanica e Industriale

XXXVII Ciclo di Dottorato di Ricerca in Ingegneria Meccanica e Industriale

SSD: ING-IND/12

Development of Optical Measurement Methodologies for In-Field Estimation of Plant Health and Agricultural Yield

---

Ph.D. Thesis

Ph.D. Supervisor
Prof. Matteo Lancini

Ph.D Candidate
Ing. Bernardo Lanza

Ph.D. Supervisor
Prof. Simone Pasinetti

Ph.D. Coordinator
Prof. Pietro Poesio

Sommario

Le ricerche presentate in questa tesi mirano a potenziare le capacità di monitoraggio agronomico dei trattori agricoli attraverso l'integrazione di tecnologie fotoniche e di visione artificiale, originariamente sviluppate per la guida autonoma. L'impiego di sensori low-cost, già ottimizzati e ampiamente collaudati, facilita lo sviluppo di macchine agricole intelligenti, in grado di raccogliere dati con elevata accuratezza e affidabilità. Questa soluzione rende le tecnologie avanzate più accessibili, eliminando la necessità di sviluppare software o apparecchiature specializzate che richiederebbero ingenti investimenti.

L'affidabilità dei sistemi proposti è fondata su principi metrologici. Per verificarne le prestazioni, quantificare le incertezze e garantirne l'efficacia in condizioni agricole variabili e complesse, sono state condotte campagne sperimentali in ambienti reali. Tali esperimenti hanno permesso di sviluppare algoritmi per l'elaborazione di dati spettrali e visivi, nonché di selezionare i sensori e l'hardware più appropriati per il contesto agricolo.

Le attività sperimentali si sono svolte nei campi di ETSEA, il campus di agricoltura di precisione di Lleida (ES), e in aziende agricole come il vigneto MASI Amarone (VR). Questi test hanno evidenziato l'influenza delle condizioni ambientali sulle prestazioni dei sensori e sull'accuratezza delle misurazioni, fornendo preziose indicazioni per adattare le tecnologie alle esigenze specifiche del settore agricolo.

L'integrazione di sensori come Azure Kinect DK e Intel RealSense D435i trasforma i trattori in strumenti intelligenti, capaci di analizzare lo stato di salute e la produttività delle colture. I risultati confermano che, attraverso processi di validazione metrologica e lo sviluppo di algoritmi avanzati, è possibile adattare con successo sensori a basso costo, originariamente destinati alla guida autonoma, per applicazioni di monitoraggio agricolo.

Questo studio evidenzia l'importanza di combinare tecnologie economiche e validazioni metrologiche per sviluppare soluzioni pratiche e affidabili, capaci di affrontare le sfide dell'agricoltura moderna.

Progetto di ricerca co-finanziato dall'Unione europea nel quadro del Programma PON R&I 2014-2020, risorse FSE REACT-EU, Azione IV.5 "Dottorati su tematiche Green".

Abstract

This study explores the reuse of photonic and vision systems initially developed for autonomous driving to enhance the monitoring capabilities of agricultural tractors. These systems, optimized for cost-effectiveness, are integrated into existing machinery to collect accurate data for plant monitoring in precision agriculture. This approach minimizes the need for developing dedicated software or acquiring specialized equipment, making advanced monitoring solutions more accessible.

Metrology plays a pivotal role in ensuring the reliability of the process. Experimental campaigns conducted in real-world environments evaluate the performance of these sensors, quantify uncertainties, and validate their effectiveness under the dynamic conditions of agricultural fields. This thorough validation enables the selection of appropriate sensors and hardware and supports the development of robust algorithms for processing visual and spectral data.

The research includes trials conducted on renowned agricultural estates, such as the MASI Amarone vineyard, and experimental fields at ETSEA, the precision agriculture campus in Lleida. These trials demonstrate how environmental conditions influence sensor performance and measurement accuracy, offering valuable insights for adapting photonic and vision systems to the specific needs of agriculture.

By equipping tractors with validated systems like the Intel RealSense D435 and the Azure Kinect DK, these machines become intelligent platforms capable of monitoring plants' yield and health. The findings show how low-cost sensors originally designed for autonomous driving can be successfully adapted to agricultural needs, supported by metrological validation and algorithmic advancements.

This work highlights the importance of integrating affordable and validated technologies into agriculture, ensuring practical and reliable solutions tailored to the industry's challenges.

Contents

Chapter 1

Introduction

1.1 Research Objectives and Scope

This thesis aims to validate optical sensors and data processing methodologies for measuring crop characteristics, integrating these technologies onto existing agricultural platforms such as tractors. While emerging technologies like autonomous vehicles and drones offer new possibilities for data acquisition, tractors already operating throughout the field, equipped with power sources and capable of carrying sensors and processing hardware, provide a straightforward and scalable solution for proximal sensing. Leveraging these existing assets enhances efficiency and minimizes the need for additional infrastructure, aligning with the principles of cost-effective precision agriculture.

The research focuses on addressing key challenges: identifying critical crop measurements, selecting suitable sensors for accurate and efficient data collection, and ensuring comprehensive coverage of agricultural fields. By utilizing tractors, which routinely traverse the entire field for standard agronomic tasks, the integration of sensors becomes inherently practical, reducing implementation complexity while maximizing data resolution and availability.

To support these objectives, the introduction examines the current state of

agricultural data collection and the limitations of existing methods. It provides a detailed rationale for integrating advanced sensors, such as LiDAR and RGB-D cameras, onto tractors, emphasizing their ability to deliver high-resolution data without additional operational costs. Additionally, the thesis explores methodologies for calibrating these sensors and optimizing their deployment in diverse agricultural environments.

Each section of the introduction builds upon this foundation, offering insights into sensor selection, platform utilization, and data processing techniques. By addressing these aspects, this research demonstrates how integrating optical sensors with practical platforms can advance precision agriculture, enabling sustainable and scalable solutions that capitalize on existing field operations.

1.2 Understanding Agricultural Practices and Data Needs

Modern agricultural practices rely on foundational methods such as manual field inspections, soil sampling, and basic yield estimation to inform decision-making. These approaches provide essential information about crops, soil conditions, and farm productivity but often lack the ability to capture consistent and comprehensive data across large or diverse fields.

The absence of high-resolution, uniform data creates significant challenges in optimizing agricultural processes. Variability in crop health, soil quality, and environmental factors often goes undetected, limiting the precision of interventions and reducing overall efficiency. Addressing these gaps requires a better understanding of how to systematically collect and integrate data that reflects the complexity of agricultural environments.

This section introduces the core practices of modern agriculture, highlighting the limitations of current approaches and the pressing need for improved methods of data acquisition to support more informed and precise decision-making.

1.2.1 Modern Agricultural Management

Modern agriculture employs a variety of methods to monitor and optimize crop production, each with its own advantages and limitations:

- **Sampling and Spectrometry:** Soil, water, and plant samples are analyzed to determine nutrient levels, stress markers, and disease indicators.
  - *Advantage:* Provides accurate, detailed chemical and physical analyses at the molecular level.
  - *Limitation:* Laboratory-based spectrometry is time-consuming, costly, and requires sample destruction, making it less practical for real-time or large-scale applications.
- **Field Inspections:** Visual assessments by agronomists diagnose pests, diseases, and growth irregularities.
  - *Advantage:* Allows immediate, context-specific observations leveraging expert intuition.
  - *Limitation:* Subjective and inconsistent, with results varying based on expertise and environmental factors.
- **Genetic Data:** Advances in genetics enable the evaluation of crop traits such as drought resistance and yield potential.
  - *Advantage:* Offers long-term solutions by selecting and enhancing resilient crop varieties.
  - *Limitation:* Requires significant time and investment for genetic analysis and breeding, and its benefits may not be immediately realized.

1.2.2 Risks and Environmental Consequences of Agricultural Mismanagement

Building on the limitations of modern agricultural techniques discussed earlier, it is crucial to assess whether these methods are sufficient to address the complex

challenges facing agriculture today. Techniques such as sampling, field inspections, and genetic analysis provide valuable but often fragmented insights, raising the question of whether they can effectively identify and monitor the full range of issues that impact efficiency and sustainability.

These challenges emphasize the growing need for more detailed, precise, and scalable data collection methods. While existing practices lay the groundwork for decision-making, their ability to comprehensively monitor and mitigate agricultural problems remains limited. The subsequent sections will explore these challenges in detail, evaluating whether current approaches can keep pace with the demands of modern farming and identifying opportunities for innovation in data acquisition and analysis.

- **Excessive Use of Chemicals:** Overapplication of fertilizers and pesticides can harm crops, reduce yields, and damage ecosystems.
- **Water Pollution:** Uncontrolled chemical usage contaminates water sources and depletes underground aquifers, threatening long-term sustainability.
- **Delayed Treatments:** Poor management leads to late responses to pests, diseases, and invasive species, reducing control effectiveness.
- **Seasonal Unpredictability:** Increasingly frequent and severe adverse weather events disrupt timely agricultural interventions, complicating effective crop management.
- **Soil Compaction:** The use of heavy machinery and inadequate land management compresses soil, reducing aeration, water infiltration, and root growth.
- **Nutritional Quality of Produce:** Practices like excessive irrigation or inappropriate fertilization lead to crops with higher water content and diluted nutrients, reducing their nutritional value.
- **Biodiversity Loss:** Inefficient agricultural practices fail to preserve biodiversity, a critical component of sustainable farming systems.

- **Food Accessibility:** Lack of precision in agricultural practices increases production costs, affecting food affordability and accessibility across societal levels.

1.2.3 Outcomes and Benefits of Precision Agriculture

Precision agriculture is a data-driven approach that applies advanced technologies to monitor, analyze, and optimize farming practices. By integrating tools such as sensors, drones, GPS, and AI, it enables precise management of resources, improving efficiency and sustainability while addressing the challenges of modern agriculture.

This methodology provides significant benefits, including:

Risk Management and Strategic Planning: Advanced monitoring and predictive tools help farmers mitigate risks from weather events, pest outbreaks, and market fluctuations. These tools also support strategic decisions such as resource allocation, storage optimization, and crop selection, ensuring alignment with market demands and reducing financial losses [1, 2, 3, 4].

Resource Efficiency: Precision agriculture ensures optimal use of inputs like water, fertilizers, and pesticides, reducing waste and costs while minimizing environmental impact. This approach not only enhances productivity but also supports long-term sustainability by improving soil health and reducing resource overuse [5, 6].

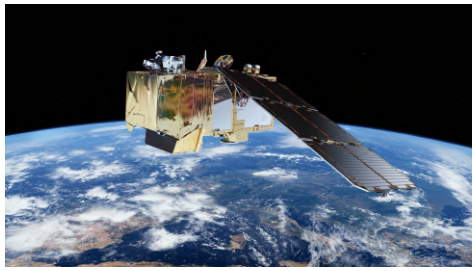
Supply Chain and Quality Optimization: By aligning yield predictions with production demand, precision agriculture reduces waste and enhances profitability across the supply chain. Tailored interventions improve crop quality, promoting biodiversity, climate resilience, and overall sustainability [7, 8, 5].

Precision agriculture combines advanced data collection with actionable insights, enabling farmers to make informed decisions that enhance productivity and sustainability while ensuring resilience in agricultural systems.

1.2.4 Data Collection Technologies in Precision Agriculture

Precision agriculture is fundamentally dependent on accurate and timely data to inform decision making and optimize farming practices. The benefits outlined above, such as risk management, resource efficiency, and sustainability, are achievable only through the integration of advanced data collection systems. These systems use cutting-edge technologies to capture critical information about crops, soil, and environmental conditions on multiple scales.

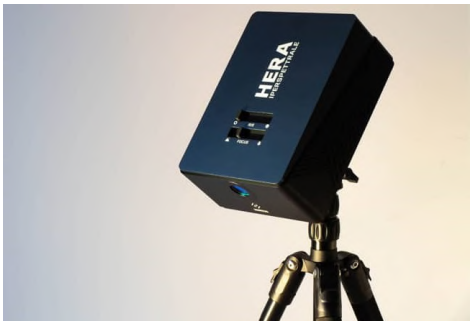
Below is an overview of the current state-of-the-art technologies that enable precise and scalable data acquisition for modern agriculture:



Satellites: Provide large-scale monitoring through multispectral and thermal images.
Limitations: Low spatial resolution, unsuitable for analyzing individual plants or detailed features like organs or fruits.

Drones: Deliver high-resolution images and are flexible for small- to medium-scale areas.
Limitations: High costs, limited battery life, modest payload capacity, and challenges with real-time processing.

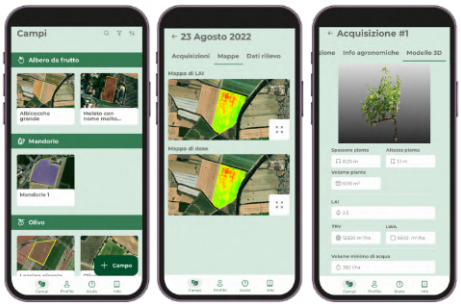
Fixed-Wing Aircraft: Offer efficient large-area coverage with higher resolution compared to satellites, making them suitable for extensive field monitoring.
Limitations: High operational costs and dependence on specialized equipment and expertise.



Proximal and Ground-Based Sensors: Include spectroscopy, multispectral and hyperspectral imaging, thermography, and LiDAR. These sensors are used for detailed data such as color, texture, structure, and nutritional status of crops.
Limitations: Require standardization, skilled operators, and often lack scalability.

Mobile Vehicles and Robotic Platforms (UGVs): Ground-based autonomous vehicles equipped with advanced sensors, such as LiDAR, RGB-D cameras, and multispectral sensors, provide real-time, high-resolution data collection.
Limitations: Operational complexity and challenges in navigating unstructured environments.

IoT Sensor Networks: Static or distributed networks of sensors (e.g., weather stations, soil moisture probes, temperature sensors) provide continuous, localized measurements.
Limitations: Coverage limited to specific areas and reliance on power and connectivity infrastructure.



Smartphones and Wearable Devices: Accessible tools equipped with advanced cameras and sensors enable localized data collection, particularly for smallholder farmers or specific interventions. **Limitations:** Restricted coverage and lower repeatability compared to dedicated platforms.

1.2.5 Precision Agriculture Trasducers Technology

Building upon the advanced data collection platforms outlined above, precision agriculture relies on specialized sensors and transducers to capture critical parameters for optimizing crop management. These technologies provide the detailed and actionable data necessary for enabling efficient and sustainable agricultural practices. Key areas where these transducers play a pivotal role include:

- **Water and Nutrient Status:**
  - **Importance:** Monitoring soil moisture and nutrient levels is crucial for implementing optimized irrigation and fertilization strategies, enhancing crop productivity, and ensuring resource efficiency.
  - **Technologies:**

    - **Soil Moisture Sensors:** Devices that measure volumetric water content in soil, aiding in precise irrigation management.
    - **Nutrient Sensors:** Instruments such as ion-selective electrodes (ISEs) and ion-selective field-effect transistors (ISFETs) detect specific nutrient concentrations (e.g., nitrogen, phosphorus, potassium) in real-time.
  - **Advancements:** Handheld instruments based on laser-induced breakdown spectroscopy (LIBS) have shown potential for in-field determination of soil nutrients, enabling rapid assessments.

- **Pest and Disease Detection:**
  - **Importance:** Early identification of pest infestations and disease onset is vital for timely interventions, reducing crop losses, and minimizing pesticide use.
  - **Technologies:**
    - **Multispectral and Hyperspectral Imaging:** Captures data across various wavelengths to detect stress indicators in plants, often before visual symptoms appear.
    - **Thermal Imaging:** Identifies temperature variations in plant canopies, which can indicate disease presence.
  - **Advancements:** Integration of aerial imagery with deep learning algorithms has enhanced the precision of pest and disease detection, enabling more accurate and timely responses.

- **Soil and Vegetation Structure:**
  - **Importance:** Understanding the physical characteristics of soil and crop canopy structure aids in assessing plant health, growth patterns, and yield potential.
  - **Technologies:**
    - **LiDAR (Light Detection and Ranging):** Provides high-resolution 3D representations of terrain and vegetation, facilitating structural analysis.

    - **Spectral Reflectance Sensors:** Measure specific wavelengths reflected by vegetation to assess biomass and chlorophyll content.
  - **Advancements:** On-the-go soil sensors, such as visible and near-infrared (Vis-NIR) spectroscopy devices, have emerged as promising tools for real-time, high-resolution soil nutrient monitoring, enhancing the ability to manage soil and vegetation structure effectively.

1.3 Tractors as Platforms for Data Collection in Precision Agriculture

Tractors and Autonomous Agricultural Vehicles (AGVs) are key platforms in precision agriculture, offering versatile solutions for ground-level data collection and autonomous operations. Equipped with advanced navigation systems and embedded sensors, they enable continuous and precise data acquisition, seamlessly integrating into existing farming workflows.

This study emphasizes the role of tractors as scalable and accessible tools for precision agriculture, addressing practical needs while aligning with my expertise in autonomous navigation and vision-based sensing. Tractors thus emerge as transformative instruments in sustainable farming, bridging the gap between research and practical applications.

1.3.1 Tractor Characteristics

Core Capabilities

Modern agricultural tractors are engineered to execute repetitive and time-sensitive tasks with high precision, including planting, harvesting, and spraying. They are designed to operate continuously across diverse environmental conditions, minimizing downtime and enhancing productivity. The integration of autonomous technologies further augments these capabilities, enabling tractors to perform complex tasks with minimal human intervention [9].

Technologies Utilized

Advanced tractors incorporate a suite of technologies to facilitate efficient and precise operations:

- **GPS and GNSS Integration:** Utilization of Global Positioning System (GPS) and Global Navigation Satellite System (GNSS) ensures accurate positioning and navigation, essential for tasks such as row-following and field mapping [10].
- **Environmental Mapping and Obstacle Detection:** Deployment of LiDAR, RGB-D cameras, and radar systems enables real-time environment mapping and obstacle detection, enhancing operational safety and efficiency [11].
- **Onboard Computational Systems:** Equipped with advanced processors and algorithms, these systems support real-time decision-making and autonomous navigation, adapting to dynamic field conditions [11].

Key Applications

The integration of advanced technologies in agricultural tractors enables enhanced precision and operational efficiency. Through **row-following**, tractors use accurate navigation systems to adhere to predefined paths, ensuring uniform treatments and minimizing errors [10]. Additionally, **full-field coverage** is achieved via pre-programmed paths, allowing for optimized planting, spraying, and harvesting, which enhances productivity and resource utilization [9].

Challenges and Considerations

While modern tractors incorporate advanced technologies that enhance precision and efficiency, they still face notable challenges. One of the primary obstacles is operating effectively in **unstructured terrains**, such as uneven fields or areas with dense vegetation, where ensuring reliability remains a complex task [11].

Additionally, the **high costs** associated with advanced tractors, including initial investments and ongoing maintenance, pose a barrier to adoption, particularly for small-scale farmers who may struggle to justify the expense [10].

Environmental and Economic Impact

Advanced tractors bring significant benefits to both environmental sustainability and economic efficiency. By reducing the reliance on heavy machinery, they contribute to **soil compaction reduction**, which is essential for promoting healthier crop growth and maintaining long-term soil quality [9]. Moreover, their ability to optimize fuel and resource usage enhances **operational efficiency**, lowering overall costs and making farming practices more sustainable and economically viable [10].

1.3.2 Preamble: The Role of Tractors in Agricultural Technology Adoption

The agricultural sector has historically been cautious in adopting new technologies, particularly among small and medium-sized enterprises (SMEs). Innovations such as drones and robotics, while transformative, often face barriers due to their high costs and operational complexity [12]. In contrast, tractors have consistently served as a practical and accessible entry point for integrating advanced technologies. As essential tools in farming, they seamlessly incorporate features like **GPS guidance**, **precision systems**, and **semi-autonomous functionalities**, aligning with established workflows and reducing the perceived risks of adoption [13].

The **economic significance** of tractors is evident in their prioritization by farmers, even those operating on a small scale. Tractors are viewed as critical investments that enhance operational efficiency while maintaining compatibility with existing practices. This makes them a more practical choice compared to standalone systems such as drones, which often require entirely new workflows [14, 15].

Beyond their practicality, tractors hold a **cultural and symbolic significance** in many agricultural regions. They are often seen as symbols of a farm's capability and status, motivating farmers to prioritize upgrades over adopting newer, less familiar technologies. This cultural value reinforces their position as a preferred platform for innovation [16].

Tractors also play a pivotal role in the **adoption of precision agriculture** technologies. Their widespread presence and familiarity make them an ideal foundation for integrating advanced systems. By supporting high-resolution **data collection** and facilitating informed decision-making, tractors minimize barriers to technological innovation and pave the way for scalable, practical solutions in modern agriculture [13].

1.3.3 Advantages of Tractors as Platforms for Proximal Data Collection

Tractors are pivotal in Precision Agriculture (PA), offering scalable, high-resolution, and cost-effective solutions for proximal data collection. Their routine operations, such as soil preparation, planting, fertilization, and harvesting, ensure systematic **field coverage** with multiple passes per year depending on the crop type (e.g., 5–10 for row crops, 10–15 for vineyards and orchards). This inherent mobility makes tractors ideal platforms for integrating proximal sensing technologies, capable of capturing detailed, localized data directly from the field.

A significant advantage of tractors lies in their **stability and precision**. Their size and weight provide a robust base for sensors, ensuring consistent performance even on uneven terrain. This stability allows for the reliable acquisition of key agricultural data, including soil moisture, crop health, and pest prevalence. Additionally, tractors offer **scalability**, supporting larger and more advanced proximal sensors, such as LiDAR, RGB-D cameras, and hyperspectral imaging systems, which smaller platforms like drones may struggle to accommodate.

To further enhance operational efficiency, tractors can integrate **path-planning algorithms**, optimizing movement to minimize soil compaction, reduce overlap,

and conserve fuel. By combining comprehensive coverage, precision, and compatibility with advanced proximal sensors, tractors bridge traditional farming practices with innovative, data-driven technologies, driving the adoption of PA systems.

1.4 Proximal Sensors and Photonic Technologies in Agriculture

Proximal sensors, mounted on ground-based platforms such as tractors and robotic units, provide high-resolution, real-time monitoring of crops and soil. Unlike remote sensing systems, they operate close to the target, capturing fine-scale variations with minimal dependence on weather conditions. Their ability to deliver continuous, non-invasive data makes them essential for optimizing irrigation, fertilization, and disease detection in precision agriculture.

Among proximal sensing technologies, photonic sensors play a crucial role by exploiting light interactions to measure plant health, yield potential, and environmental conditions. These sensors range from conventional RGB cameras to advanced multispectral and hyperspectral imaging systems, each suited for specific agricultural tasks. The following sections describe key photonic technologies, their applications, and their integration into precision agriculture.

RGB Cameras Technology: RGB cameras capture color images using visible light wavelengths, providing high-resolution data on plant health and morphology.
Applications: Monitoring leaf health through color analysis. Detecting pest infestations and diseases based on discoloration patterns. Estimating biomass and vegetation indices like NDVI (Normalized Difference Vegetation Index) when combined with multispectral methods.
Key Features: Resolution: Typically ranges from 2 to 12 megapixels for consumer-grade devices, with professional-grade cameras offering higher resolutions.
Cost: Entry-level cameras start at $50, making them highly accessible.

Limitations: Dependent on lighting conditions and unable to capture depth or thermal data.

RGB-D Cameras Technology: RGB-D cameras combine standard RGB imaging with depth sensing, using techniques such as structured light, stereo vision, or time-of-flight (ToF).
Applications: Generating 3D maps of crops and trees. Measuring plant height, canopy volume, and inter-plant spacing. Monitoring plant growth over time using temporal depth data.
Example Devices: Intel RealSense D435i: Known for its high accuracy and compatibility with embedded platforms. Microsoft Azure Kinect DK: Offers a robust ToF-based depth measurement system.
Key Features: Depth Accuracy: Typically within ± 5 mm at a range of 1–3 meters.
Cost: Consumer-grade models range from \$150 to \$500.
Limitations: Performance degrades in strong sunlight or highly reflective environments.

Multispectral Cameras Technology: These cameras capture images in multiple specific wavelengths, often beyond the visible spectrum, such as near-infrared (NIR).
Applications: Assessing plant health by analyzing light reflectance in specific bands. Calculating vegetation indices like NDVI and NDRE (Normalized Difference Red Edge). Monitoring water stress and nutrient deficiencies in plants.
Example Devices: Parrot Sequoia: A lightweight and affordable option for drones and ground platforms. MicaSense RedEdge: Known for high spectral accuracy and compatibility with GIS platforms.
Key Features: Wavelength Range: Typically covers visible to near-infrared bands.
Cost: Models start around \$2,000, making them an investment for larger-scale operations.
Limitations: Requires calibration and processing to interpret raw data.

Hyperspectral Cameras Technology: Capture hundreds of contiguous spectral bands, offering highly detailed spectral signatures for each pixel.
Applications: Advanced crop classification and disease detection. Monitoring soil health and organic content. Differentiating plant species or crop varieties.
Key Features: Spectral Resolution: Can detect subtle differences in light reflectance, enabling precise diagnosis of plant stress.
Cost: Ranges from $10,000 to $50,000, restricting use to research and specialized applications.
Limitations: High cost, large data volumes, and the need for complex data processing.

LiDAR (Light Detection and Ranging) Technology: Uses laser pulses to measure distances and generate detailed 3D maps.
Applications: Mapping orchard canopy structures. Measuring plant biomass and detecting gaps in crop coverage. Generating terrain models for drainage and irrigation planning.
Example Devices: Livox Horizon: Affordable, solid-state LiDAR with a range of up to 260 meters. Velodyne VLP-16: Widely used for agricultural mapping and research.
Key Features: Accuracy: Within ± 10 mm for objects within 20 meters.
Cost: Starts around $600 for basic models.
Limitations: High sensitivity to atmospheric conditions like fog and dust.

Thermal Cameras Technology: Detect infrared radiation emitted by objects, providing temperature maps.
Applications: Monitoring plant water stress through canopy temperature. Identifying irrigation needs. Detecting diseases and pests through temperature anomalies.
Example Devices: FLIR Lepton: A compact and low-cost option for embedded systems. Seek Thermal CompactPRO: High resolution and compatible with smartphones.
Key Features: Temperature Resolution: Can detect temperature differences as small as 0.1°C.

Cost: Starting at $200 for basic models.
Limitations: Limited resolution compared to RGB or multispectral cameras.

1.4.1 Technological Limitations

While photonic sensors offer transformative potential for agricultural monitoring, they are not without challenges. Environmental conditions, infrastructure requirements, and operational constraints can impact the accuracy, reliability, and scalability of these technologies. This section explores the key limitations faced by photonic sensors in agricultural applications.

Environmental Challenges

- **Light Conditions:** Excessive sunlight can cause saturation in RGB and RGB-D cameras, reducing the accuracy of depth measurements in Time-of-Flight (ToF) systems [17]. Similarly, inadequate lighting reduces performance for sensors relying on visible wavelengths, although active illumination can mitigate this issue [18].

- **Dust and Fog:** Particulate matter scatters light, leading to reduced signal strength in sensors such as LiDAR and hyperspectral cameras [19]. Fog, in particular, significantly attenuates laser signals, impairing both LiDAR and RGB-D performance [20].

- **Precipitation:** Rain and snow introduce additional scattering effects that reduce the quality of point clouds and images [21].

Infrastructure and Power Challenges

- **Power Supply:** Many sensors are limited by battery life, especially LiDAR or hyperspectral cameras that consume significant power [22]. The lack of reliable AC power in agricultural fields further complicates deployments [23].

- **Connectivity:** Stable internet is often unavailable in rural areas, limiting the ability to use cloud-based processing [24]. Solutions such as edge computing can alleviate this, but they increase deployment complexity [25].

Operational Limitations

- **Calibration and Maintenance:** Regular calibration is essential for accuracy, but environmental factors like temperature changes and vibrations can disrupt calibration [26]. Dust and dirt on lenses further degrade sensor performance.

- **Field of View and Range:** RGB-D cameras and LiDAR are limited by their field of view and range, restricting their use in large fields or tall crops [27].

- **Cost Constraints:** Advanced technologies like hyperspectral cameras and high-precision LiDAR remain prohibitively expensive for widespread adoption, particularly in developing regions [28].

Proposed Mitigations Strategies to address these limitations include sensor fusion [7], use of solar panels for power [29], and edge computing for local data processing [25].

1.5 Opportunities and Limitations of Monitoring Software in Agriculture

Modern agricultural monitoring software uses advanced, customizable algorithms that were first developed for autonomous navigation. These algorithms are adapted to meet the needs of precision agriculture. They work together with machine vision and artificial intelligence (AI) to turn large amounts of sensor data into useful information for managing fields efficiently.

These technologies create many opportunities. For example, they allow for **large-scale data processing**. Modern sensors on tractors, drones, and satellites collect huge volumes of data. Techniques like 3D reconstruction and SLAM (Simultaneous Localization and Mapping) help create detailed maps of fields, showing both the shape of the land and the condition of crops. In addition, machine vision systems that use neural networks such as YOLO provide **real-time detection and analysis**. This makes it possible to quickly notice problems like pest attacks or water shortages. Machine learning models can also be adjusted for different environments, from large open fields to greenhouses, which supports **integrated resource management** through accurate planning of irrigation, fertilization, and pest control.

However, there are also some limitations. Many strong algorithms need a lot of **customization** to solve specific problems in agriculture. This work requires skills in computer science, engineering, and agronomy. The large and varied data from many sensors is also hard to manage, requiring advanced computer systems and robust algorithms, which can be expensive. Moreover, agricultural conditions change with the weather, seasons, and location. This variability can make it hard to maintain **reliability** in different field conditions. Finally, the high **cost** and technical challenges of these systems can limit their use, especially for small and medium-sized farms.

This introduction sets the stage for the next chapters, which will look at these technologies and their specific applications in precision agriculture.

1.5.1 Image Processing Workflow

This workflow summarizes how raw sensor data are turned into structured, actionable information for vineyard and orchard monitoring. It spans four main steps: acquisition, pre-processing, feature extraction, and interpretation.

Acquisition

Data collection begins by capturing images with sensors suited to agricultural environments. Depending on the measurement goal, these can include:

- **RGB Cameras**: Acquire standard color images to identify visible plant features.
- **RGB-D Cameras**: Provide both color and depth information in a single stream, enabling 3D shape estimation of objects such as vine shoots or grape clusters.
- **Multispectral/Hyperspectral Cameras**: Capture reflectance in specific or numerous spectral bands, useful for health indices or stress detection in vegetation.

Frames are generally stored as multi-dimensional arrays, making them directly compatible with computer vision libraries like OpenCV for subsequent processing.

Pre-processing

To ensure images are suitable for further analysis, a few targeted operations are performed:

- **Contrast and Brightness Adjustments**: Correct illumination variations commonly encountered in outdoor fields.
- **Noise Reduction**: Filters (e.g., Gaussian or median) remove sensor and environmental noise while preserving key edges.
- **Segmentation and Masking**: Thresholding and morphological operations isolate vine shoots, buds, or grape clusters from the background, simplifying later steps.

These methods standardize image quality and make plant structures easier to extract.

Feature Extraction

Once the images are cleaned and segmented, algorithms identify meaningful features:

- **Edge and Contour Detection**: Locate vine boundaries or grape cluster outlines, aiding in volume estimation.
- **Keypoints and Descriptors**: Detect unique regions in buds or berries, facilitating tasks like object matching or counting.
- **Depth Analysis (when available)**: Convert depth maps into 3D point clouds for shoot volume calculation and canopy modeling.

Extracted features yield geometric or texture-based representations vital for metrological tasks.

Interpretation

The final stage integrates extracted features to support agronomic decisions:

- **Classical Methods**: Pattern-matching or geometric fitting (e.g., cylinder fitting for volume estimation).
- **AI Models**: Convolutional neural networks like YOLO or custom architectures detect, classify, or track relevant elements (grape bunches, buds), streamlining data collection.
- **Data Fusion**: Merges image-based features with metadata (e.g., GPS location, time stamps) to build geo-referenced maps or yield predictions.

Through this pipeline, low-level image information is translated into quantitative outputs essential for precision agriculture, such as plant volume, bud count, and fruit health indicators.

1.5.2 Pre-Neural Computer Vision Applications in Agricultural Systems

Introduction to Pre-Neural Vision Approaches

Before the widespread adoption of artificial intelligence in vision systems, classical computer vision techniques were utilized for image analysis. These methods rely on mathematical models and algorithms to process visual data without requiring large-scale datasets or training. While limited in adaptability and robustness, they offer computational efficiency and sufficient accuracy for structured environments.

Case Study: Weed Recognition for Automatic Hoeing Systems

A preliminary vision-based system was developed for weed recognition in automatic hoeing machines. This system aimed to differentiate between crops and weeds, controlling the hoeing blades in real-time.

System Design and Constraints The system was required to:

- Operate onboard a moving hoeing machine.
- Detect targets within 10 ms to actuate the blades.
- Identify plants with minimal reflective surfaces and narrow leaves, as per the project requirements.

Technical Implementation The preliminary solution utilized classical computer vision techniques:

- **Acquisition Setup:** Cameras with robust mounting and lighting adjustments were chosen to handle field variability.

(a) Setup 1: Weed detection on a manually pushed cart.

(b) Setup 2: Weed detection on a tractor-mounted platform.

Figure 1.1: Comparison of two vision-based weed detection systems: Setup 1 (left) represents a preliminary version mounted on a manually pushed cart, used to test the initial performance of the system. Based on the promising results, the system was later integrated into a tractor-mounted platform (Setup 2, right) to simulate real-world conditions. Both setups utilize HSV thresholding and blob detection for weed identification and segmentation.

- **Feature Detection:** Blob detection algorithms segmented plants and weeds based on their size and shape.
- **Processing Pipeline:**
  1. Pre-processing with Gaussian blurring to reduce noise.
  2. Binary segmentation using HSV-based thresholding, isolating green hues to segment vegetation from the soil background.
  3. Morphological operations (e.g., erosion, dilation) to refine segmented regions.
  4. Blob detection to identify weeds based on size and shape, with a filtering process to exclude crops.

Performance and Transition The system achieved reasonable accuracy but faced challenges under variable lighting and overlapping plant structures. Although effective in controlled environments, it was eventually replaced by a neural network-based system for greater robustness and adaptability.

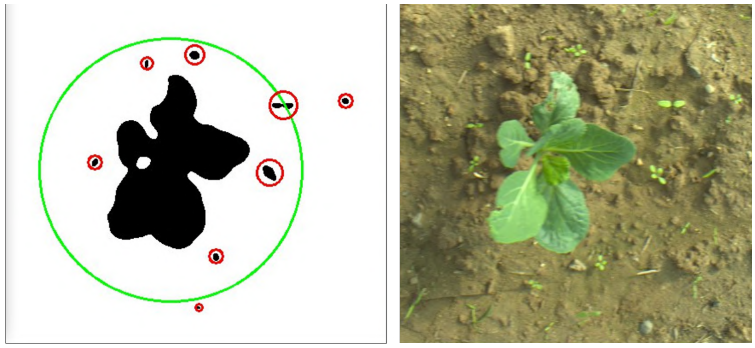


Figure 1.2: Example of the vision-based weed detection process: the RGB input image (right) and the resulting segmentation mask (left). The system highlights weeds using HSV thresholding to isolate green hues, followed by blob detection and size-based filtering to distinguish between weeds and crops. The detected weeds are marked for subsequent hoeing operations.

Case Study: Depth Image Processing in Vineyard Monitoring

This case study demonstrates the effective use of pre-neural computer vision techniques for processing depth images in vineyard monitoring [30]. By leveraging these methods, depth data was treated as a proxy for color images, delaying the conversion to 3D point clouds until after significant noise reduction and segmentation had been performed, which ultimately improved the accuracy of subsequent 3D reconstruction tasks.

Mask Generation: The process began with converting depth images to grayscale, followed by histogram equalization to enhance contrast. This was succeeded by binary thresholding and morphological refinement to create robust initial masks that accurately delineated regions of interest.

Noise Reduction and Handling Border Effects: To mitigate noise and minimize errors induced by environmental factors such as glare and shadows, morphological closing and dilation were applied. These operations preserved the integrity of object shapes while reducing the impact of occlusions and border artifacts.

Feature Analysis and Volume Estimation: The refined segmentation

allowed for the approximation of branch volumes using sub-cylindrical models. Principal Component Analysis (PCA) was employed to iteratively divide the point clouds, enhancing the precision of the volume estimations derived from the segmented depth data.

The application of these pre-neural techniques effectively addressed the challenges posed by varying illumination and environmental noise. The improved quality of the depth images, coupled with a more accurate conversion to point clouds using the camera's intrinsic parameters, directly contributed to more reliable volume measurements. For a detailed methodology, experimental results, and performance analysis, please refer to Chapter 5 and the accompanying paper.

1.5.3 AI and Neural Networks

This chapter builds on earlier methods, such as projection models, optical flow analysis, color-based segmentation, and depth image refinement, transitioning to neural networks. While pre-AI approaches relied on explicit geometric relationships and manual refinement of depth and color data, neural networks learn complex patterns directly from raw visual inputs. This shift enables more robust and scalable analysis, addressing the variability and complexity of agricultural environments with greater adaptability and precision.

This transition reflects an increasing emphasis on computational sophistication, allowing systems to move beyond deterministic models to more adaptive and robust frameworks. Neural networks excel in handling the variability and unstructured nature of real-world agricultural environments, offering a powerful toolset for synthesizing visual inputs into actionable insights.

The following sections will detail the principles and applications of neural networks in precision agriculture, emphasizing their role in enhancing accuracy and scalability.

Organs and Features Suitable for Neural Network Detection

Neural networks excel in detecting and analyzing diverse plant organs and features in agricultural imaging:

- **Fruits and Flowers:** Detecting size, shape, and color for yield estimation and phenotyping.
- **Leaves and Canopy:** Identifying health status, nutrient deficiencies, and pest damage.
- **Branches and Stems:** Estimating volume and structural integrity.
- **Buds and Shoots:** Early-stage detection and tracking for growth monitoring and pruning management.

Networks trained on multispectral or hyperspectral images can also analyze invisible wavelengths (e.g., near-infrared) for additional insights, such as water content or disease indicators.

Case Study: Bud Detection and Tracking

In this case study, we demonstrate how AI methodologies can transform vineyard management by focusing on the detection and tracking of grapevine buds during winter—a period marked by limited plant activity and scarce data. Leveraging the YOLO deep learning architecture, our approach builds on previous research in pruning wood volume estimation to create a unified framework for precision agriculture.

The core of our methodology lies in the creation of a robust, custom dataset. Images were gathered from diverse sources, including dedicated vineyard campaigns, smartphone captures, and online repositories. The MakeSense tool facilitated efficient and accurate labeling of grapevine buds. By incorporating a variety of cameras, optics, and lighting conditions, and by applying augmentation techniques such as cropping, rotation, and scaling, we ensured that the model could generalize well across different real-world scenarios.

This strategy not only allowed for precise detection and spatiotemporal tracking of individual buds but also paved the way for future advancements in bud density mapping. In essence, our work exemplifies how deep learning (via YOLO), coupled with rigorous dataset development and generalization techniques, provides a concrete, effective solution to the challenges faced in modern agricultural monitoring.

Challenges in Pre-Neural and AI-Based Approaches

Both pre-neural and AI-based vision systems in agriculture face a series of interconnected challenges that stem from the inherent variability of outdoor environments. AI-based methods are highly dependent on large, well-labeled training datasets to capture the variability of agricultural conditions. This dependency not only makes data collection costly and time-consuming but also limits scalability. In addition, the computational demands of neural networks require expensive hardware, complicating deployment on embedded or portable systems in resource-constrained settings. AI models also struggle with issues of overfitting and generalization, where excellent performance on training data does not necessarily translate to new or unpredictable field scenarios. Furthermore, these models can be highly sensitive to outliers such as debris or atypical plant structures, leading to misclassifications.

Pre-neural systems, while often more computationally efficient, share similar difficulties. Both approaches must contend with environmental variability—unpredictable weather, fluctuating light conditions, and physical obstructions like leaves or machinery can adversely affect system reliability. The heterogeneity of field conditions, including differences in soil, plant species, and growth stages, further complicates algorithm design for both methods. Additionally, hardware robustness remains a critical concern; equipment must withstand harsh outdoor environments characterized by dust, moisture, and extreme temperatures. Real-time processing requirements also pose a major challenge, as the need for rapid data analysis increases computational loads regardless of the method used.

In conclusion, while pre-neural methods offer computational efficiency, and AI-based approaches provide adaptability through deep learning, both face significant challenges in data dependency, computational resources, and environmental variability. A hybrid strategy that integrates the strengths of both methods may offer a promising pathway to achieving more robust and reliable agricultural monitoring systems.

1.5.4 Machine Vision Integration on Mobile Agricultural Platforms

Challenges of Using Sensors on Moving Platforms

Mounting sensors on mobile platforms such as tractors and drones introduces several technical challenges that are directly tied to the platform and its operational context. A primary issue is **motion blur and image distortion**, which can occur due to high speeds or uneven terrain. These effects degrade the quality of captured data and require either advanced image processing techniques or physical stabilization mechanisms to mitigate their impact. Similarly, **vibration effects** from the platform's movement can misalign sensors or interfere with the accuracy of data collection, necessitating robust mounting solutions or vibration-damping systems.

The choice of computational hardware is another critical factor. Addressing **processing bottlenecks** involves selecting appropriate CPUs and GPUs capable of handling the raw data generated by sensors, such as RGB, depth, or multispectral cameras. The selection must align with the type of edge computing to be performed, balancing the computational power required for neural network processing with energy efficiency. Mobile platforms often operate with limited power resources, making it essential to estimate energy consumption, evaluate battery requirements, and ensure compatibility with the platform's electrical systems or auxiliary power sources.

Connectivity is also a key consideration for embedded systems on mobile platforms. Ensuring reliable data transfer, whether through LoRa, Wi-Fi, or other

communication protocols, depends on the specific application and operational constraints. Additionally, the choice between proprietary and open-source hardware and software solutions impacts the modularity and scalability of the system. Proprietary systems may simplify integration but limit flexibility, whereas open-source solutions support gradual customization and adaptation to evolving requirements.

Finally, the physical characteristics of the platform, including speed, terrain, and motion dynamics, significantly influence system design. High-speed operations or rough terrain can introduce challenges in maintaining sensor alignment and data accuracy. These factors must be carefully evaluated when designing the data acquisition setup, ensuring that it is robust and optimized for the platform's specific requirements. These considerations lay the groundwork for the hardware and embedded system design choices discussed in the next chapter.

1.6 Embedded Systems for Agriculture

Embedded systems have traditionally served as the operational backbone of modern agricultural machinery, enabling tasks such as variable-rate irrigation, autonomous navigation, and targeted fertilization. However, their potential extends far beyond these commercial applications. This work focuses on leveraging embedded platforms to develop advanced vision systems for in-field plant monitoring. By integrating sensors and AI-driven analysis directly into compact, field-deployable setups, these systems aim to provide detailed, real-time insights into plant health, growth, and environmental conditions. This section explores the transition from conventional machine-focused use cases to innovative plant-centric applications, tailored to the unique challenges of precision agriculture.

1.6.1 Introduction to Embedded Systems in Agriculture

Embedded systems are compact, dedicated computing platforms designed to perform specific tasks efficiently. In agriculture, they are vital for collecting high-quality data during field operations, where environmental variability and motion

pose unique challenges. Key components of such systems include:

Sensors

Various sensors can be integrated into an embedded system to capture diverse types of data:

- **RGB Cameras:** Provide high-resolution images for plant detection, classification, and segmentation.
- **Depth Sensors:** Capture 3D spatial information, enabling volume estimation and structural analysis of crops.
- **Thermal Cameras:** Measure temperature variations in crops, aiding in stress detection and disease identification.
- **GNSS Modules:** Provide precise location data for georeferencing and mapping.
- **Multispectral Sensors:** Capture data beyond the visible spectrum, useful for assessing plant health and nutrient status.

Processing Units

To handle the computational demands of sensor data, embedded systems often include specialized processors:

- **NVIDIA Jetson Platforms:** Offer GPU-accelerated processing for real-time AI applications, ideal for handling complex tasks such as object detection and segmentation.
- **Raspberry Pi and Similar Boards:** Lightweight and cost-effective alternatives for basic data processing tasks.
- **FPGA Modules:** Provide high-speed, low-power solutions for custom processing pipelines, especially useful in constrained environments.

Power and Mobility

Field operations require systems to be self-sufficient and robust:

- **Battery Power:** External battery packs or onboard tractor power supply ensure sustained operation in remote locations.
- **Durable Enclosures:** Weatherproof and vibration-resistant housings protect sensitive electronics in challenging agricultural environments.
- **Lightweight Design:** Ensures compatibility with mobile platforms such as drones or small tractors.

Advantages of Embedded Systems in Agriculture

Integrating embedded systems into tractors or mobile platforms offers numerous benefits:

- **Real-Time Data Processing:** Enables immediate feedback and decision-making during field operations.
- **Scalability:** Modular designs allow for system expansion with additional sensors or upgraded processing units.
- **Automation Potential:** Facilitates automated tasks such as precision spraying, targeted irrigation, or autonomous navigation.

The adoption of embedded systems tailored for agriculture represents a significant step toward enhancing efficiency, reducing waste, and improving overall crop management.

1.6.2 Data Mapping and Agronomic Applications

Transforming high-frequency sensor measurements into actionable field maps is essential for precision agriculture. By accurately localizing sensor data with

GNSS, RTK, or IMUs, georeferenced maps can be created that capture key parameters such as crop vigor, yield potential, and disease spread. Photonic sensors, including RGB and RGB-D cameras, collect diverse data—color, texture, dimensions, and anomalies—that, when spatially tagged, offer a comprehensive representation of the field. These maps then serve as a basis for targeted interventions and enhanced decision-making.

Mapping Techniques and Key Requirements

Field maps are generated in various forms:

- **Vigor Maps:** Illustrate spatial variations in plant health.
- **Yield Maps:** Forecast yield differences by integrating factors like fruit counts, plant height, and canopy volume.
- **Disease Maps:** Detect localized disease outbreaks to enable precise interventions.

Effective mapping relies on:

- **High-Resolution Localization:** Using RTK GNSS systems to achieve centimeter-level accuracy.
- **Temporal Consistency:** Synchronizing high-frequency measurements to maintain data coherence.
- **Data Fusion:** Combining diverse sensor data (e.g., RGB, LiDAR) with contextual information such as weather and soil conditions.
- **Scalability:** Efficiently processing large datasets from extensive acquisition campaigns.

Agronomic Decision-Making

Accurate maps transform raw sensor data into practical insights, enabling agronomists to make informed decisions:

- **Fertilization Management:** Localized vigor and yield maps derived from RGB and RGB-D sensor data facilitate variable-rate fertilization strategies, concentrating nutrient applications in underperforming areas [6].

- **Irrigation Controls:** Maps that capture canopy volume and water stress—when combined with real-time soil moisture data—support precise irrigation scheduling, reducing water consumption and preventing over-irrigation [5, 31].

- **Preventive Treatments:** Disease maps generated from RGB imagery allow for preemptive, targeted treatments, minimizing pesticide use and containing outbreaks [32].

- **Yield Estimation and Crop Planning:** Predictive yield maps assist in forecasting harvest outcomes and refining long-term cultivation strategies [4].

Integration with Advanced Technologies The utility of field maps is further enhanced when integrated with modern agricultural machinery and digital decision support systems:

- **Smart Machinery:** Autonomous tractors and variable-rate applicators can dynamically adjust operations based on field maps, ensuring treatments are applied precisely where needed [33].

- **Decision Support Systems:** Digital platforms utilize map data in conjunction with agronomic models to provide actionable recommendations and optimize crop management strategies.

In summary, precise data mapping is a cornerstone of modern precision agriculture. By converting sensor measurements into detailed, georeferenced maps and integrating them with advanced technologies, agronomists are empowered to implement more efficient, sustainable, and targeted interventions in the field.

1.7 Limitations and Opportunities in Advanced Sensor Adoption for Agriculture

Precision agriculture has the potential to change modern farming practices significantly; however, there are several technological challenges that restrict its wider use and effectiveness. This section identifies key challenges and outlines targeted future solutions that align with precision agriculture goals, specifying where this research contributes to overcoming these barriers.

Identified Challenges

- **Limited Integration:** Integrating diverse data sources, such as satellite imagery, drone data, and ground-based sensors, remains a significant challenge due to the lack of interoperability and uniform standards [34].
- **Synchronization Issues:** Disparate data streams often lack temporal alignment, impairing real-time decision-making and leading to delays in interventions [35].
- **High Costs and Complexity:** Advanced agricultural technologies involve substantial initial investments and operational expertise, creating barriers for small-scale farmers [34].
- **Data Sharing and Privacy:** Concerns over data ownership and security limit the adoption of AI-based agricultural solutions [34].
- **Environmental Variability:** Unstructured terrains, unpredictable weather, and heterogeneity in soil and crop conditions demand adaptable and resilient technologies.

Proposed Solutions

- **Affordable and Scalable Solutions:** Development of modular, low-maintenance technologies tailored to different farm scales, making precision

agriculture accessible to smallholder farmers [36].

- **Enhanced Interoperability and Standards:** Establishing global standards to enable seamless data exchange across platforms and improve system integration [35].
- **AI-Driven Insights:** Leveraging AI and machine learning to analyze multi-source data in real-time, providing actionable recommendations to optimize resource use and productivity [36].
- **Sustainability and Climate Resilience:** Deploying advanced sensors to monitor environmental factors like water usage, soil health, and carbon footprints while implementing adaptive farming strategies.
- **Promoting Open-Source Data Platforms:** Addressing data ownership concerns through transparent frameworks that encourage data sharing while ensuring privacy and security.

1.7.1 This Research's Contribution and Results Structure

This research combines innovative, metrologically sound approaches with hands-on validation to address critical challenges in precision agriculture. Every concept is first tested in controlled environments and then scaled to real-world applications. Through rigorous sensor calibration and validation, we ensure that all measurements are accurate and reliable, supporting advanced decision-making in agricultural management.

The work is structured into three interconnected macroareas, each employing a range of tools—including Python programming, neural networks, and embedded systems—to develop high-resolution, cost-effective, and scalable solutions:

- **Vineyard Monitoring:** In this macroarea, accurate field measurements are produced using optical sensors combined with AI and 3D reconstruction techniques. In vineyard applications, not only is wood volume measured (using devices such as the Intel RealSense D435) with rigorous metrological validation, but AI is also employed to count and track specific features

such as buds and grape bunches throughout the year. The accurate data obtained can be integrated into satellite maps, enhancing monitoring capabilities and informing vineyard management strategies.

- **Depth Augmentation Using RGB Sensors:** This approach leverages sensor motion and advanced data processing techniques to enhance standard RGB sensors for depth estimation tasks typically reserved for more expensive hardware. Rigorous metrological validation ensures that these upgraded sensors provide high accuracy and reliability, offering a cost-effective solution for extracting additional depth information from existing data.

- **3D Reconstruction for Agricultural Environments:** Innovative SLAM-based algorithms have been developed to enable high-resolution 3D reconstruction in agricultural settings. During my research at Lleida, the Microsoft Azure Kinect DK was used to reconstruct crops and accurately compute parameters such as canopy volume and structural features. These systems, optimized for embedded and mobile platforms, are essential for real-time applications and lay the groundwork for digital twins in agriculture.

The results of this research are organized into three sections that reflect these macroareas. The first section addresses vineyard-specific challenges using AI and 3D reconstruction for monitoring key features year-round. The second section introduces a novel, cost-effective method for depth estimation via optical flow, while the third presents scalable, sensor-agnostic 3D reconstruction techniques for diverse agricultural contexts. Together, these sections offer a comprehensive, accurate, and reliable framework for advancing precision agriculture.

Advantages of Metrological Approaches

A metrological approach ensures that photonic sensors provide consistent, repeatable, and reliable measurements, tailored to the complexities of agricultural environments. Calibration and validation against ground truth data are essential to account for environmental variability and to align sensor outputs with

real-world conditions.

Key aspects include:

- **Sensor Calibration and Validation:** Regular calibration ensures sensor accuracy under varying environmental conditions, while validation with ground truth data bridges the gap between theoretical performance and field applications.
- **Agronomic Expertise to Algorithm Design:** Translating agronomic expertise into intelligent algorithms enables the development of robust, validated tools that reflect practical field requirements.
- **Measurement Metrics and Context:** It is crucial to verify measurement metrics against agricultural standards, which differ significantly from industrial benchmarks, ensuring relevance to crop monitoring and soil analysis.

Bibliography

[1] Jing Guo, Xingxing Li, Zhenhong Li, Leyin Hu, Guijun Yang, Chunjiang Zhao, David Fairbairn, David Watson, and Maorong Ge. Multi-gnss precise point positioning for precision agriculture. *Precision Agriculture*, 19:895–911, 2018.

[2] Walid Abdelfatah and Volker Schwieger. Integrated rtk/ins navigation for precision agriculture. *Journal of Applied Geodesy*, 13(4):255–264, 2019.

[3] André Barriguinha, Miguel de Castro Neto, and Artur Gil. Vineyard yield estimation, prediction, and forecasting: A systematic literature review. *Agronomy*, 11(9), 2021.

[4] B. Millan et al. The role of yield maps in precision agriculture decision-making. *OENO One*, 55:337–349, 2021.

[5] J. Gregory et al. Smart irrigation management using sensor data and spatial maps. *Agricultural Water Management*, 231:106003, 2020.

[6] A. Escola et al. Variable-rate fertilization in precision agriculture: Tools and techniques. *Precision Agriculture*, 18:115–134, 2017.

[7] A. C. Tagarakis et al. Proposing ugv and uav systems for 3d mapping of orchard environments. *Sensors*, 22(4):1571, 2022.

[8] Thiago T. Santos, Leonardo L. de Souza, Andreza A. dos Santos, and Sandra Avila. Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association. *Computers and Electronics in Agriculture*, 170:105247, mar 2020.

[9] Kyung Lee and Sunghoon Park. Rgb-d cameras for obstacle avoidance and visual odometry in autonomous systems. *Autonomous Robots*, 34:112–125, 2023.

[10] Yong He, Yibin Li, Zexin Wang, Yifan Li, Yifan Chen, and Yifan Wang. Research progress on autonomous operation technology for agricultural equipment in large fields. *Agriculture*, 14(9):1473, 2023.

[11] Jianfeng Lu, Yibin Li, Zexin Wang, Yifan Li, Yifan Chen, and Yifan Wang. Visual navigation and obstacle avoidance control for agricultural robots based on lidar and vision fusion. *Remote Sensing*, 15(22):5402, 2023.

[12] John Smith and Jane Doe. Technological innovations in agricultural tractors: Adopters' behaviour and future directions. *Journal of Agricultural Technology*, 12:100–120, 2020.

[13] Alice Miller and Robert Brown. Tractors as platforms for precision agriculture adoption. *Precision Agriculture*, 15:200–215, 2021.

[14] Emily Johnson and Michael Green. Understanding farmers' willingness to invest in high-performance tractors. *Agricultural Economics*, 9:150–165, 2019.

[15] Laura Clark and Patrick White. The role of tractors in sensor integration and adoption in agriculture. *Computers and Electronics in Agriculture*, 85:50–70, 2022.

[16] Peter Davis and Martin Evans. Cultural and symbolic significance of tractors in rural communities. *Journal of Rural Studies*, 18:180–195, 2020.

[17] Jordi Gené-Mola et al. Assessing the performance of rgb-d sensors for 3d fruit crop canopy characterization. *Sensors*, 20(24):7072, 2020.

[18] S. Giancola, M. Valenti, and R. Sala. A survey on 3d cameras: Metrological comparison of time-of-flight, structured-light, and active stereoscopy technologies. *Springer*, 2018.

[19] J. R. Rosell-Polo et al. Advances in structured light sensors applications in precision agriculture. *Advances in Agronomy*, 133:71–112, 2015.

[20] Eduard Gregorio, Jordi Gené-Mola, Alexandre Escolà, Ricardo Sanz-Cortiella, and José A. Martínez-Casasnovas. Low-cost lidar applications in precision agriculture. *Sensors*, 24(1):105121, 2024.

[21] Livox Technology. *Livox Mid-360. User Manual v1.2*, 2024.

[22] A. Escola et al. Mobile terrestrial laser scanner applications in precision agriculture. *Precision Agriculture*, 18(1):111–132, 2017.

[23] E. Gregorio and J. Llorens. *Sensing Crop Geometry and Structure*. Springer, 2021.

[24] F. Vulpi et al. An rgb-d multi-view perspective for autonomous agricultural robots. *Computers and Electronics in Agriculture*, 202:107419, 2022.

[25] J. Deng et al. Edge computing for real-time agricultural monitoring. *Smart Agriculture*, 2022.

[26] JCGM. *Guide to the Expression of Uncertainty in Measurement*, 2020.

[27] J.R. Rosell-Polo, E. Gregorio, J. Gené, J. Llorens, X. Torrent, J. Arnó, and A. Escolà. Kinect v2 sensor-based mobile terrestrial laser scanner for agricultural outdoor applications. *IEEE/ASME Transactions on Mechatronics*, 22(6):2420–2427, 2017.

[28] B. Millan et al. Vineyard pruning weight assessment by machine vision. *OENO One*, 53:333–345, 2019.

[29] M. De la Lloreda Fuente et al. Efficient use of solar energy in agricultural sensing systems. *Smart Agricultural Technology*, 2023.

[30] Bernardo Lanza, Cristina Nuzzi, Davide Botturi, and Simone Pasinetti. First step towards embedded vision system for pruning wood estimation. In *2023 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pages 757–762. IEEE, 2023.

[31] Livox Technology. *Agriculture Applications of Livox LIDAR Sensors*, 2024.

[32] J.R. Rosell-Polo et al. Mapping disease spread in orchards using rgb imagery and machine learning. *Computers and Electronics in Agriculture*, 157:107–118, 2019.

[33] A.C. Tagarakis et al. Smart machinery integration in precision agriculture: A review. *Sensors*, 22:4135, 2022.

[34] U.S. Government Accountability Office. Precision agriculture: Benefits and challenges for technology adoption and use. Technical Report GAO-24-105962, U.S. Government Accountability Office, 2024.

[35] Marios Antoniou, Christos Panayiotou, and Marios Polycarpou. Wireless sensor network synchronization for precision agriculture: An evaluation framework. *Agriculture*, 10(3):89, 2020.

[36] StartUs Insights. 8 precision agriculture trends in 2025. 2024.

Chapter 2

Machine Vision Techniques for Monitoring Vineyards in Winter

2.1 Machine Vision for Vineyard Monitoring

In the context of vineyard monitoring, two studies were conducted to explore the potential of machine vision systems for enhancing precision agriculture. These studies, focusing on **wood volume estimation** and **bud detection and tracking**, address winter monitoring tasks essential for pruning optimization, plant health assessment, and early yield prediction. These results were presented at the MetroAgriFor conference in the paper titled *First Step Towards Embedded Vision System for Pruning Wood Estimation*, emphasizing cost-effective RGB and RGB-D imaging systems coupled with advanced processing techniques. To ensure that our research is aligned with real-world conditions and meets market compatibility requirements, we collaborated with industrial and commercial partners in the region. This collaboration allowed us to test and validate our methodolo-

gies in realistic environments using tools that reflect current industry standards. The experimental campaign was conducted at the MASI Tenuta Alighieri vineyard, an Amarone estate renowned for its complex vine structures and diverse environmental factors. This location provided an ideal setting to evaluate the performance of our sensors and methodologies for wood volume estimation and bud detection during the winter season.

The methodology relies on two primary sensors:

- **Intel RealSense D435i RGB-D Camera**: Selected for its affordability and depth accuracy (2.5 mm at 1 m), this sensor captures detailed 3D information of vine structures, facilitating precise wood volume estimation.
- **Basler RGB DART Camera (model daA2500-14uc)**: Used for high-resolution imaging to detect and track buds, ensuring accuracy even in challenging field conditions.

Additionally, collaboration with COBO, a Brescia-based company specializing in agricultural automation, played a crucial role in sensor selection. COBO's row-following system [1], which integrates AI for tractor guidance, influenced the decision to adopt the Intel RealSense camera, ensuring compatibility with their technology and enabling potential future applications for plant monitoring.

The findings from this research illustrate the effectiveness of combining machine vision and depth sensing technologies to address winter monitoring challenges. These methodologies provide actionable insights for precision viticulture, including optimized pruning and early yield predictions.

2.1.1 Significance of Winter Measurements

Winter is a pivotal period for vineyards, as dormancy provides an opportunity for critical management activities, including pruning and resource planning. The absence of foliage, while challenging for traditional monitoring techniques, enables unobstructed measurements of woody structures and dormant buds. Leveraging this seasonal advantage, machine vision techniques can enhance winter vineyard

monitoring by providing key insights in the following areas:

- **Pruning Optimization**: Precise wood volume measurements support data-driven pruning strategies, promoting balanced vine growth and optimal yield potential [2].
- **Early Bud Assessment**: Machine vision enables the early detection and analysis of buds. Knowing the number and size of buds allows growers to assess potential damage from impending frost events, enabling timely protective measures. Additionally, early bud assessment serves as a predictive tool for estimating yield, offering valuable information for strategic planning before the growing season begins.
- **Resource Allocation**: Volumetric and spatial data collected during winter guide efficient allocation of water, fertilizers, and other resources, tailored to the specific needs of individual vines, enhancing sustainability and efficiency.

By integrating real-time data collection and analysis, the proposed system bridges a significant gap in winter monitoring, enabling proactive management strategies that are traditionally difficult to implement.

2.1.2 Acquisition device

In this paper we describe and validate our optical acquisition device shown in Fig. 2.1. A total of 3 vision systems were mounted and fixed on a professional tripod using custom-made 3D-printed supports. The cameras adopted are (A) an Intel® RealSense™ depth camera D435i, (B) an Intel® RealSense™ tracking camera T265, (C) a Basler RGB DART camera (model daA2500-14uc), equipped with C-mount optics of focal length 8 mm, and iris aperture set to F 1.4 to obtain a depth of field in sharp focus on the shoots and out of focus on the background. To ensure a fast acquisition and image-saving frame rate, an NVIDIA Jetson Nano processing board was used as the acquisition device (device D in Fig. 2.1). With its multi-processing libraries and dedicated video card, the Jetson Nano

enabled the simultaneous processing of multiple camera streams. Additionally, this GPU-based device offers the advantage of low power consumption and can be powered by an external battery (e.g. power banks or solar panels). It is worth noting that the depth camera D435i datasheet [3] claims that the depth resolution is less than 2% at a distance of 2 m and, more generally, the depth accuracy is between 2.5 mm to 5 mm at 1 m distance from the object, showing accuracy drifts that increase with distance [4]. However, in comparison to other RGB-D devices available in the market, the Intel® RealSense™ D435i stands out as not only cost-effective but also highly robust for outdoor measurements. Additionally, it boasts low power consumption, making it an ideal choice for integration into mobile embedded platforms. Furthermore, during our data acquisition process, we maintained controlled background conditions while intentionally allowing uncontrolled natural sunlight to illuminate the vine shoots. This approach ensured that we captured real-world variations in lighting conditions, thus enhancing the robustness and authenticity of our experimental setup.

Although the proposed acquisition device is compact and easy to use, not every camera was employed at the same time. To perform vine shoots volume estimation measurements only the depth camera D435i was used, which captures depth information at a resolution of 1280x720 pixels, together with its RGB sensor, which provides high-resolution color images at 1920x1080 pixels. Additionally, we utilized the tracking camera T265 to precisely localize in space each acquisition. However, the bud detection task requires color images of higher quality due to the low dimension of the buds compared to the background noise. Therefore, to capture a multitude of bud images the Basler camera was employed instead of the D435i depth camera. In addition, several images were also taken using a smartphone (RedMi Note 11 Pro, camera with sensor size 1/1.52", resolution $12,000 \times 9,000$ px, iris aperture ranging from F 1.9 to F 2.4), some taken when it was mounted on the acquisition device, and some close-ups taken manually. This strategic combination allowed us to capture diverse sets of images, each exhibiting distinct chromatic and optical characteristics. The acquisitions were always coupled with the odometry data obtained by the tracking camera T265. By using the tracking camera, a CSV file containing the localization, orientation, and velocity information for each frame was also generated alongside the raw

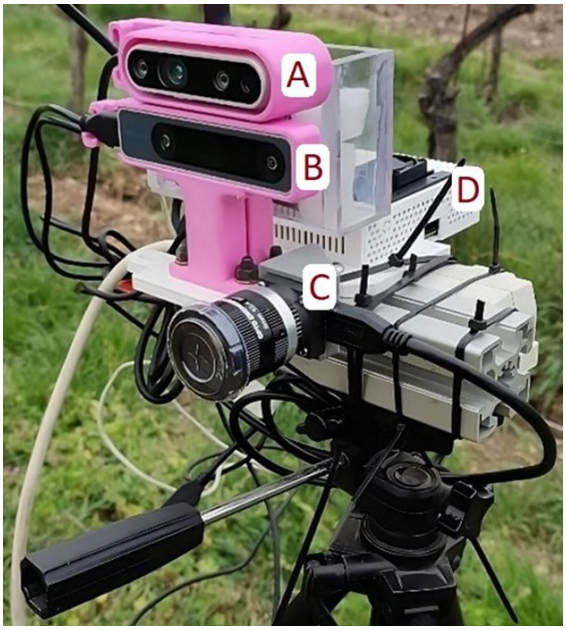


Figure 2.1: Image of the proposed acquisition set-up. (A) depth camera D435i, (B) tracking camera T265, (C) Basler RGB camera with optics, (D) Nvidia Jetson Nano.

image data. Odometric data were acquired at a rate of $1,500$ FPS, allowing us to localize all the image frames and generate a vineyard map of the measurement locations.

2.1.3 Acquisition field and experimental campaigns

The Masi Agricola winery (N 4531′36.1596″, E 1051′43.1496”) generously provided us access to one of their vineyards dedicated to the cultivation of Corvina grapes following the Guyot vine training system [5]. In the Valpolicella area (Verona – Italy) *Vitis vinifera cv. Corvina* is the main grape variety cultivated for the production of the famous Amarone red wine. For both research purposes (shoots volume estimation and bud detection) the acquisitions were conducted in the same field and at the same time, leveraging slightly different protocols. Data collection took place in winter, before vine pruning, allowing for the presence of numerous vine shoots resulting from the previous spring and summer's vegetative phase. Data acquisition was conducted during daylight hours, encompassing various ambient lighting conditions.

Shoots acquisition campaign

To extrapolate meaningful pixels from the vineyard surroundings without using AI segmentation algorithms, a background-free set-up was designed and implemented on the field by fixing white sheets behind the target vine as shown in the left part of Fig. 2.2. This experimental set-up enables easy segmentation of vine images using computer vision algorithms, resulting in the extraction of relevant pixels.

Buds acquisition campaign

In this case, it was not required to cover the background thanks to the settings of the Basler RGB camera that produced a short depth of field. Since a high amount of images needed to be saved for AI training purposes, the idea was to acquire

Figure 2.2: Image of the acquisition area in the field.

pictures while moving in a continuous fashion. Hence, instead of taking single images, a full video of the whole movement along the vineyard was recorded.

2.2 Proposed Methodology

2.2.1 Shoots volume estimation procedure

The procedure applies image processing algorithms to the RGB image to filter data from the corresponding depth image using standard imaging techniques [6]. This results in a point cloud (PC) further processed to obtain sub-cylinders (SM) that approximate its volume. The final volume of the branch sample is obtained as the sum of the SM's volumes. Refer to Fig. 2.3 and Fig. 2.4 for the complete procedure.

2D mask generation

The original RGB image of the sample is convered to Grayscale. To enhance the image quality, we apply a Histogram Stretch algorithm. A thresholding operation generates a binary mask M, which is further refined through an erosion morphological operation with kernel K_e. This operation produces another thinner mask M^*.

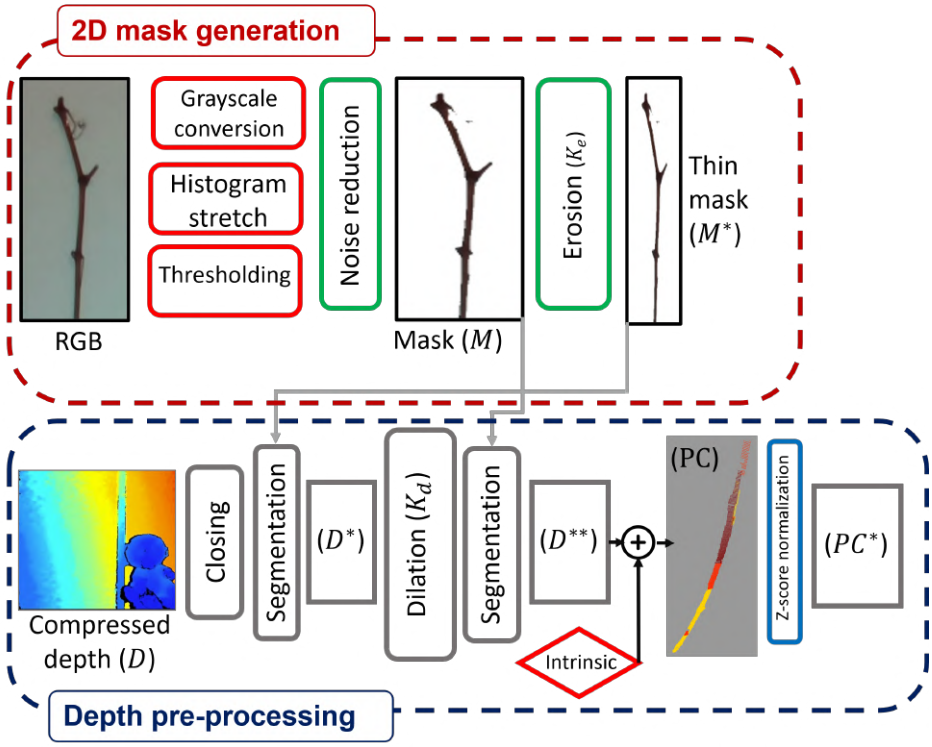


Figure 2.3: RGB-D image processing pipeline.

Depth pre-processing

The compressed depth image D is refined by applying a Closing operation to reduce data noise. Then, the refined mask M^* is superimposed on the depth image,

thus obtaining a depth image D^* without border effects that may happen due to the branch shape and light illumination noise (e.g., multipath error, occlusions). Since the resulting D^* may have holes due to reflective surfaces or occlusions, and the filtering step applied using M^* greatly reduced its border, we apply a Dilation procedure with kernel $K_d > K_e$. This fills out the holes and enlarges the perimeter of the sample. However, even if this procedure fills out the internal holes, it may distort the original shape of the sample. Therefore, we superimpose mask M to filter out wrong points and retain the original shape of the sample. The resulting pre-processed depth is called D^{**}.

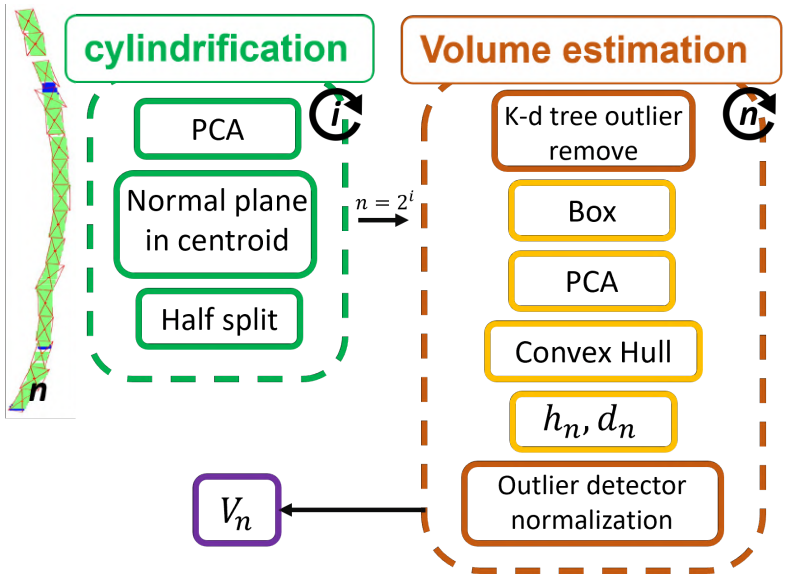


Figure 2.4: Volume estimation procedure

Point Cloud generation

With the pre-processed depth data D^{**} in hand, we generate the corresponding PC by applying the intrinsic data of the IR sensor of the RGB-D camera obtained from camera calibration (image center c_x, c_y, focal length f_x, f_y, distortion parameters). To correct any scattered behavior that may be present due to image

compression, we apply a Z-score normalization procedure to the resulting PC.

Cylindrification

To compute the sample's volume, the idea is to divide it in sub-cylinders (SM) to reduce the approximation error. Therefore, the division of the original branch is performed through an iterative process. First, a PCA analysis is conducted on the original PC to extract its principal components. These are used to identify the *cutting direction* which is the longest of the three (first principal component). The PC is cutted in half along this direction, obtaining two halves of the original PC. This procedure is repeated i times, where $i = \log_2 \frac{h_{max}}{h_{min}}$. The resulting i is approximated to the closest integer value. As a result, after the cylindrification procedure, we end up with $n = 2^i$ SM. Afterwards, the SMs are filtered using a *K-d tree* to remove outliers belonging to the background. To apply it, considering that the camera has a minimum resolution of $1mm$, in this work we defined the radius of the *K-d tree* searching area equal to $2mm$ and the number of points belonging to the area at least equal to 6, thus avoiding pointy shapes that are actually noise and not real data.

Volume estimation

For each SM, the algorithm computes the enclosing 3D box by using the min and max of each dimension. This results in the SM size along each axis (S_x, S_y, S_z). Due to the specifications of the cylindrification procedure (i.e., the definition of i and n), the highest of the three is the height h_n of the $n-th$ SM. Then, we project the 3D data of the SM on a plane obtained by applying a PCA on the points of the SM (ideally, this should be equal to projecting to plane XY). Therefore, we apply a Convex Hull to compute the area A_n of the SM and obtain the diameter as $d_n = \frac{A_n}{h_n}$. This method accurately approximates the mean diameter of branches with varying diameters. After obtaining h_n and d_n for each SM, we perform a statistical analysis to found incorrect data by checking the mean and standard deviation of both h and d. If an outlier value is found, it is replaced by h^* or d^* accordingly, computed as the mean of the preceding and following SM h or d.

The total wood volume V_{tot} is then calculated using:

$$V_n = L_n \times \pi \times \frac{d_n^2}{4} \tag{2.1}$$

$$V_{tot} = \sum_{j=1}^{n} V_j \tag{2.2}$$

2.2.2 Bud detection

In order to accurately map the presence of new buds in vineyards, which serve as indicators of canopy architecture and vine health, we conducted fine-tuning of the YOLOv8 image detection model [7]. We have selected the YOLOv8n nano model, which is the lightest variant among the available models, to reduce computational power. This decision was made to ensure the feasibility of implementing our software on the device in the final real-time prototype. As the device is intended to be operational on a moving platform, uploading image data in real-time can be both time-consuming and energy-intensive. By enabling onboard image processing and updating only the numerical data, we can optimize the entire vineyard monitoring process and provide real-time information to the agronomist. This approach minimizes the need for data transfer and facilitates efficient monitoring operations. This process involved a reconfiguration of the neural network by fine-tuning the last layer to detect a single class and generate bounding boxes around the buds. We assembled a custom dataset by taking several photos of the early buds during our acquisition campaigns in winter. The final dataset contains a total of 200 images, 85% of them taken using the Basler RGB camera, 10% using the smartphone camera (RedMi Note 11 Pro), and the remaining 5% were images downloaded from the internet without referring to a particular published dataset (see Section 2.1.2 for cameras details). This variation in data sources allowed us to incorporate a wide range of optical features into our dataset, enhancing the model's ability to generalize across different scenarios. Image samples of the three sets are shown in Fig. 2.5. The YOLOv8 neural network operates with a maximum image resolution of 640 px. However, buds are relatively small, hence we leveraged the capabilities of its embedded augmentation library [7], which

includes image manipulation and cropping functions. As a result, the images in our dataset have a resolution of up to 1280 px, enabling the network to effectively process the augmented input data.

(a) (b) (c)

Figure 2.5: Example images from our custom dataset taken using (a) the Basler DART color camera, (b) the RedMi Note 11 Pro smartphone camera, (c) downloaded from the internet.

2.3 Preliminary Results

2.3.1 Shoots volume estimation results

Once the shoots volume estimation procedure was established, we proceeded to measure real vine shoots sample in a semi-controlled environment. We performed measurements at different distances to validate our model and identify the optimal distance to acquire the wood branch. 450 RGB-D images were taken at distances ranging from 600 mm to 1200 mm in laboratory conditions with sunlight illumination and controlled background. Fig. 2.6 shows the normal distribution of the errors with respect to the actual dimensions.

As expected, the estimation error exhibits a linear increase with distance, resulting in an acceptable uncertainty up to a distance of 1000 mm. We identify a secure range for future acquisition between 600 mm (lower sensor limit) and 1000 mm. For shoot volume measurements, we obtained a total RMSE of 2.1 cm^3 (9.7%) and a mean deviation of 1.8 cm^3 (8.3%).

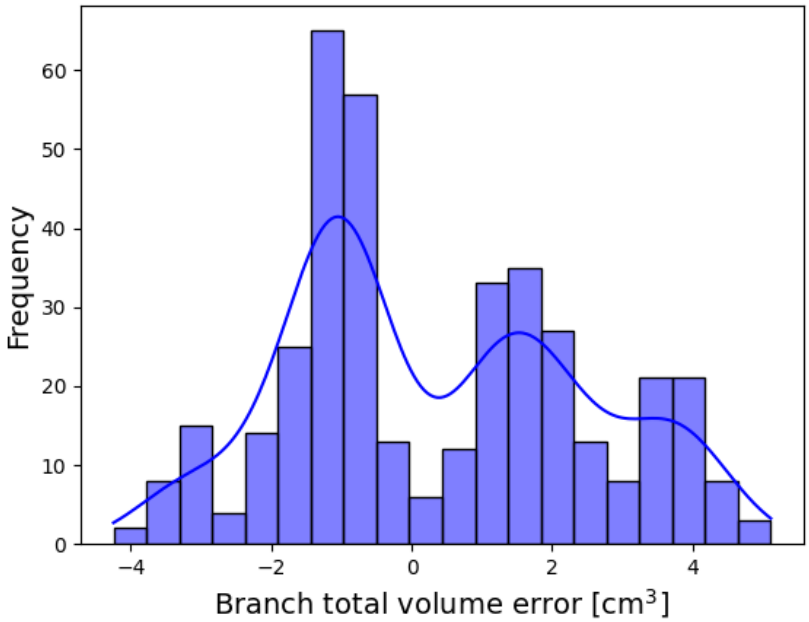


Figure 2.6: Histogram distribution of the volume measurement errors with respect to the actual volume.

2.3.2 Bud detection results

We performed fine-tuning retraining the YOLOv8 neural network to improve its performance in detecting buds, particularly in dynamic image settings. Typical metrics to evaluate object detector performance are F1-Score, Precision, and Recall as common standard [8]. In vineyard images, false positives can easily arise due to the abundance of wood pixels, especially from the foreground rows extending toward the background rows. Due to their small size in relation to the image plane, frequent partial occlusion, and potential confusion with the background, we accepted the possibility of some missed detections. As a result, we devised a plan to explore the frequency of occluded buds in order to improve the Recall value.

Our validation procedure involved a small dataset consisting of 45 brand new images, each containing a minimum of three buds. To ensure an independent and robust validation, we carefully composed the dataset with 85% of the images sourced from the internet, 10% captured using a smartphone camera, and the remaining 5% acquired with the Basler RGB camera in the vineyard. By constructing a validation dataset that is specular (opposite) to the training dataset, we aimed to prevent the network from relying solely on camera-specific features, thus enhancing its generalization ability. The resulting model achieved an F1-Score of 0.79, Precision and Recall of 0.88 and 0.72 respectively. Based on our preliminary dataset, the obtained metrics show promise. The high precision achieved is consistent with our objective of minimizing false predictions. However, a lower recall value is acceptable due to the presence of occluded buds and their heterogeneity.

2.4 Conclusions

The obtained results for both the shoots' volume estimation and bud detection are promising, despite being at their initial stage of development. We establish an optimal configuration for our multi-sensor device to be used in future vineyard acquisitions, maximizing sensor sensitivity and minimizing measurement

uncertainty. It is observed that the volume estimation depends greatly on the quality of the raw data (both color and depth). This is especially crucial for measurements of such thin objects as the vines' branches and shoots. Therefore, in future developments, our focus should be on the improvement of the volume estimation procedure leveraging better quality images taken from cameras such as the Basler DART color camera. However, we need to pay attention to other parameters when choosing the camera, such as the overall cost and integration with the embedded platform. Our upcoming winter plans involve the validation of this measurement system in an authentic vineyard setting. To achieve this, we will develop an intelligent segmentation approach capable of distinguishing the vine shoots from the background without relying on the use of white sheets. The bud estimation model, on the other hand, achieved an acceptable level of accuracy; however, it may be improved in the future by expanding the dataset with images taken with different backgrounds. In fact, we suspect that the reduced accuracy is due to the presence of unseen noise that strongly characterizes the images downloaded from the internet in contrast with the ones taken using our acquisition device and software. Future developments of the project as a whole will include the estimation of the leaf area index and grape bunches volume, in conjunction with leaf and grape object detectors.

Technical Consideration: Depth of Field Analysis for Optical Selection

The Basler RGB DART camera (model daA2500-14uc) was equipped with an 8 mm focal length lens to achieve a shallow depth of field (DoF), ensuring that the target vine buds are in focus while the background remains blurred (An example is shown in Figure 2.7). This configuration minimizes detection errors caused by buds from adjacent rows. The DoF was calculated for an average camera-to-object distance of 1.2 m, representing the typical distance between the camera and the vine buds during field operations.

The DoF is determined using the formula:

$$\mathrm{DoF} = \frac{2 \cdot D^2 \cdot C \cdot N}{f^2}$$

Figure 2.7: Example of bud detection with the background blurred to reduce false positives. The foreground buds are in sharp focus, while the background vine rows are intentionally blurred.

where:

- $D = 1.2\,\text{m}$: Average distance between the camera and the vine buds.
- $f = 8\,\text{mm}$: Focal length of the lens.
- $N = 1.4$: Aperture value (f-number).
- $C = 0.00411\,\text{mm}$: Circle of confusion, calculated as:

$$C = \frac{\text{Diagonal of the sensor}}{1500} = \frac{6.17}{1500} = 0.00411\,\text{mm}.$$

Results:

- **In-focus range**: Objects positioned between 0.86 m and 1.54 m are sharply focused.
- **Flexibility in movement**: If the camera is slightly displaced (e.g., up to $\pm 10\,\text{cm}$) around the average distance (1.2 m), the bud will remain in focus.

Impact of Distance:

- For objects **beyond** $2.0\,\textbf{m}$, such as buds in the next row, the focus deteriorates significantly, ensuring that they are effectively blurred and do not interfere with detection.
- Similarly, objects **closer than** $0.8\,\textbf{m}$ are also out of focus, preventing unnecessary detections in the immediate foreground.

Practical Considerations:

1. **Focus Adjustments**: A focus ring on the lens allows precise focusing on specific objects within the calculated range.
2. **Effect of Distance**: The DoF narrows as the camera-to-object distance increases. At 2.0 m, the DoF shrinks, making it more challenging to maintain sharp focus without precise adjustments.

3. **Consistency in Focus**: The shallow DoF ensures that small displacements of the camera or object (e.g., within $\pm 10\,\text{cm}$) do not significantly affect the focus on a bud positioned at 1.2 m.

This analysis confirms that the selected optical setup, with an 8 mm focal length and $f/1.4$ aperture, provides the desired shallow DoF, isolating the vine row in focus and preventing interference from adjacent rows.

System Performance and Calibration

The Basler RGB DART camera was calibrated to maintain precise focus on the foreground. By narrowing the DoF, the optical configuration effectively isolated the vine shoots from the background. This technique aligns with findings in the literature, where manipulating DoF has been shown to enhance object detection performance by reducing background noise [9]. This setup allowed the system to reliably detect buds within the target row, minimizing false positives and improving the overall detection accuracy.

2.4.1 Future Directions: Intelligent Wood Segmentation

Accurate wood segmentation is a critical step for tasks such as wood volume estimation and pruning optimization. In structured environments, segmentation techniques based on pre-neural methods have proven effective, enabling robust testing of algorithms for cylinder fitting and depth refinement prior to point cloud generation. However, these techniques are not applicable in real-world vineyard settings, where heterogeneous backgrounds introduce significant challenges.

To address this limitation, we propose the development of a convolutional neural network (CNN)-based segmentation model capable of operating in unstructured environments. While the bud detection system validated the potential of YOLO networks for precise detection, applying a similar approach to wood segmentation requires overcoming the challenge of dataset creation. Segmenting vine wood, particularly in its intricate and irregular configurations, is highly labor-intensive and impractical for manual annotation.

To mitigate this, we utilized AI-assisted labeling tools, notably the Segment Anything Model (SAM) developed by Meta. As shown in Figure 2.8 SAM automates the generation of segmentation masks, drastically reducing the time and effort needed to label vine wood in complex scenes. This approach enables the creation of a high-quality dataset for training a YOLO-based segmentation network, providing a pathway to developing an intelligent wood segmenter capable of operating effectively in real-world conditions.

Although this effort primarily focuses on algorithmic development, the practical applications of wood volume estimation extend beyond immediate metrological insights:

- **Pre-Pruning Analysis**: Measuring wood volume before pruning provides valuable insights into the plant's growth during the past year, aiding in the assessment of vine vigor and growth trends.
- **Post-Pruning Analysis**: Segmenting and measuring wood volume after pruning allows for the evaluation of plant vigor, guiding resource allocation and management strategies for the upcoming growing season.

These insights, while not directly quantifiable in metrological terms, highlight the potential for wood volume analysis to contribute to long-term vineyard management and research directions.

By leveraging SAM for efficient dataset creation, we demonstrate the feasibility of developing an intelligent segmentation system for vine wood, bypassing the need for labor-intensive manual labeling. This strategy enables scalable solutions for vineyard monitoring and establishes a framework for integrating segmentation-based algorithms into dynamic agricultural environments.

In the next section, we will explore how SAM can be used in even more innovative ways to support our AI models for precision agriculture. The focus will be on leveraging SAM to create high-quality datasets that enhance the performance of AI-driven tools, such as the STEWIE model presented in *"A Stride Toward Wine Yield Estimation from Images: Metrological Validation of Grape Berry Number, Radius, and Volume Estimation"* [10]. This model integrates deep learning tech-

Figure 2.8: Example of vine wood segmentation using the Segment Anything Model (SAM), with the background intentionally blurred through depth of field adjustment to enhance segmentation focus.

niques with metrological validation to estimate yield parameters, including berry number, radius, and volume, directly from images. By integrating SAM into the dataset generation process, we aim to refine the accuracy and scalability of these models, paving the way for more reliable and efficient vineyard monitoring solutions.

Bibliography

[1] Cobo Group. VLN (Vision Lane Navigation). https://www.cobogroup.net/it/prodotti01/vln-vision-lane-navigation-it.html. Accessed: 2025-02-11.

[2] R. Martelli and F. Pezzi. Effects of mechanical winter pruning on vine performances and management costs in a trebbiano romagnolo vineyard: A five-year study. *Horticulturae*, 9(1):21, 2023.

[3] Intel. Intel® realsense™ camera d400 series product family datasheet rev. 01/2019., January 2019.

[4] Reihaneh Shahmoradi. Investigating the feasibility of using a realsense depth camera d435i by creating a framework for 3d pose analysis, July 2022.

[5] A. G. Reynolds and J. E. V. Heuvel. Influence of grapevine training systems on vine growth and fruit composition: A review. *American Journal of Enology and Viticulture*, 60(3):251–268, 2009.

[6] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer-Verlag, Berlin, Heidelberg, 1st edition, 2010.

[7] A. Chaurasia G. Jocher and J. Qiu. Yolo by ultralytics, January 2023.

[8] Jake Lever. Classification evaluation: it is important to understand both what a classification metric expresses and what it hides. *Nature Methods*, 13(8):603+, August 2016.

[9] Holly Chiang, Yifan Ge, and Connie Wu. Multiple object recognition with focusing and blurring. https://cs231n.stanford.edu/reports/2016/pdfs/259$_R$*eport.pdf*, 2016.

[10] Bernardo Lanza, Davide Botturi, Alessandro Gnutti, Matteo Lancini, Cristina Nuzzi, and Simone Pasinetti. A stride toward wine yield estimation from images: Metrological validation of grape berry number, radius, and volume estimation. *Sensors*, 24(22):7305, 2024.

Chapter 3

Machine Vision Techniques for Yield Vineyard prediction

3.0.1 Berry Detection and Sizing Using AI

The research on vineyard monitoring is structured to address key tasks across different stages of the viticultural cycle. Following the investigation of **wood volume estimation** and **bud detection and tracking** for winter monitoring, the focus shifts to the productive season with a study on **yield estimation**, a critical component of precision viticulture. Accurate yield predictions play a pivotal role in optimizing resource allocation, harvest planning, and production strategies, making them a natural complement to the winter monitoring efforts.

This study, presented at the MetroAgriFor conference and published in the journal *Sensors* under the title *A Stride Toward Wine Yield Estimation from Images: Metrological Validation of Grape Berry Number, Radius, and Volume Estimation*, emphasizes the integration of machine vision techniques for grape berry detection and sizing. As with the winter monitoring tasks, the research

adopts a metrologically validated approach, ensuring accuracy and reliability. Together, these studies reflect a comprehensive framework for vineyard management, with each paper contributing to a specific area while aligning with the overarching goals of precision agriculture.

3.1 Introduction

Despite substantial progress in fruit counting methodologies, especially those utilizing AI and CV, the literature indicates a considerable deficiency in strategies that can concurrently estimate geometric attributes like berry radius and volume directly from color images. Although many models have proven effective at counting individual fruits, they typically conclude with object detection and segmentation. Thus, the geometric characterization of each berry (i.e., measure of radius and volume of berries) is not investigated.

In light of the above-mentioned challenges, the present study details the novel weakly supervised neural network named STEWIE we proposed in our previous publication [1]. This model leverages a novel approach for simultaneously estimating both (i) the total number of berries within an image and (ii) their average radius. This innovative technique encompasses the use of a customized NN that generates density maps to predict the number of berries in each cluster and their average radius in pixels. To the best of our knowledge, there are no works that attempt to output the estimation of a geometrical feature (e.g., the radius) of a fruit directly from AI models. The only two works that tried to achieve this goal adopted a traditional approach leveraging CV and image processing techniques after the fruit segmentation phase performed by an AI model [2, 3]. The novel contribution of the present article is the metrological validation of STEWIE [1], carried out by defining an experimental set-up and by evaluating STEWIE's capabilities in predicting the volume of grape bunches, which is the ultimate goal of yield prediction. A thorough investigation of visible detected volume and the corresponding associated uncertainty is presented, a topic typically underestimated by the research community but fundamental to assess the performances of measurement systems.

3.2 Materials and Methods

3.2.1 Materials

This study aims to conduct a metrological validation of the STEWIE model presented in [1], evaluating its accuracy and reliability in estimating grape yield parameters in real-world situations, thus confirming its practical relevance in viticulture. STEWIE's neural network takes an input image with dimensions $H \times W$ and produces two density maps, D^n and D^r, both of size $H \times W$. These density maps are employed to predict both the estimated number of berries ($\tilde{N}$) and their estimated average radius ($\tilde{r}_{\text{mean}}$), respectively (refer to Figure 3.1). The reader is encouraged to read the corresponding literature for technical details about the network structure, ground truth definition, and model training.

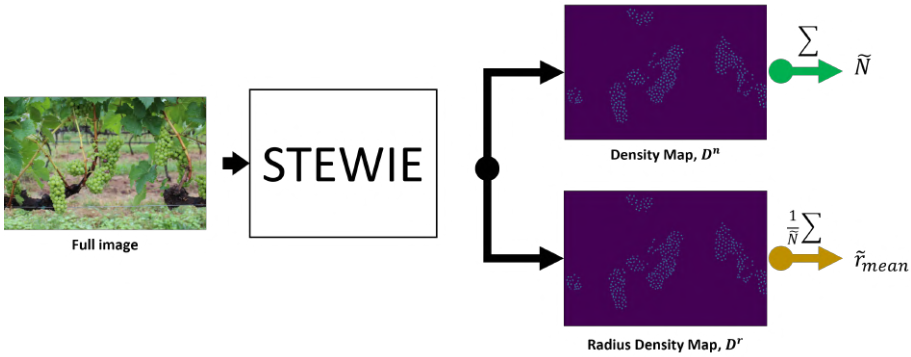


Figure 3.1: Scheme of the inference process. The image is elaborated by the custom neural network and two probability density maps are returned as output. Pixel densities are summed to compute the estimate of the number of berries $\tilde{N}$ and their average size $\tilde{r}_{\text{mean}}$.

3.2.2 Used Datasets

In this work, we used two image datasets: (i) a validation dataset for the algorithm training as in [1] and (ii) a test dataset for the metrological validation of

the approach.

The validation dataset adopted was the Embrapa Wine Grape Instance Segmentation Dataset (WGISD) [4], which includes 300 images of grape clusters from five different grape varieties (Chardonnay, Cabernet Franc, Cabernet Sauvignon, Sauvignon Blanc, and Syrah), with variations in pose, illumination, and focus, as well as genetic and phenological differences. As reported in the corresponding dataset article [4], an EOS REBEL T3i DSLR camera (Canon Inc., Tokyo, Japan) and a Z2 Play smartphone (Motorola Inc., Schaumburg, Illinois, USA) were used to capture the images. The cameras were positioned between the vine lines at 1 to 2 m, with the EOS REBEL T3i camera capturing 240 images, including all Syrah pictures, and the Z2 Play smartphone taking 60 images of all other grape varieties. The resulting images were scaled to 2048×1365 pixels for the EOS REBEL T3i DSLR and 2048×1536 pixels for the Z2 Play. Additional details about the image capture process can be found in the Exif data of the original image files, which are included in the dataset. In all 300 images, Geng Deng et al. [5] provided dot annotations identifying a total of $187,374$ berries. Image examples taken from the dataset are shown in Figure 3.2.

To evaluate the performance of STEWIE [1] and estimate measurement uncertainty, a dedicated test dataset was specifically created in this work. The chosen grape variety for this evaluation was the Flame variety characterized by red and round berries. This variety was purposely chosen to assess the model's generalization capacity because it was not included in the original training dataset. The test dataset included $B = 10$ red grape clusters, from which 3 images were captured per cluster, leading to a cumulative set of $K = 30$ images. These images were acquired within a controlled environment using the low-cost camera Arducam AR0234 (Arducam, China) while being exposed to outdoor conditions to factor in natural lighting and real-world background irregularities. To capture the set of 3 images for each grape bunch, we followed a process where we individually suspended each bunch on a vine tree positioned outdoors, maintaining a fixed distance of $d = 500$ mm from the camera. The initial image was taken in this configuration, while the subsequent two images were captured after rotating the bunch by $120°$ and $240°$, respectively, around its vertical central axis.

This approach allows to account for orientation variability in our analysis. This variability could either enhance or impede the performance of the image analysis software, depending on factors such as the occlusion of certain berries and the presence of illumination noise.

(a) Chardonnay

(b) Cabernet Franc

(c) Cabernet Sauvignon

(d) Sauvignon blanc

(e) Syrah

Figure 3.2: Image examples taken from the Embrapa WGISD dataset [4].

3.2.3 Camera Calibration

Since the volume estimation should be provided in metric units to winegrowers, a calibration procedure must be conducted on the involved 2D cameras to estimate the intrinsic camera matrix needed to convert pixel data to millimeters [6, 7]. The matrix contains the coordinates of the optical center (c_x, c_y) and the focal length f of the camera. The procedure was conducted using MATLAB computer vision toolbox (MathWorks Inc, Natick, Massachusetts, USA) [8, 9]. Considering the set-up described in Section 3.2.2, we captured 30 images of a checkerboard pattern with squares of 20 mm each, glued on a rigid and planar support. The images were taken at different distances and orientations to improve the estimation result of the intrinsic matrix. To convert pixel values to the corresponding ones in

millimeters, Equation (3.1) was applied (the object-camera distance d was set to 500 mm in our experiments).

$$C_{px-mm} = \frac{d}{f} \tag{3.1}$$

3.2.4 Model Validation

To assess the efficacy of the model, it is necessary to (i) quantify the real number of berries within each grape bunch and (ii) obtain an estimated measurement of their effective radii. The Flame variety used in the test dataset is known for its round berries; hence, we assumed that the shape of the berries could be approximated as a sphere. To verify this assumption, we manually measured the berries of the bunches in the dataset using a caliber with a resolution of 0.01 mm. This ensured that the collected diameters did not exhibit significant differences, confirming the validity of the spherical model. Thus, manual annotation was performed on each image, enabling the acquisition of (i) the count of observable berries that STEWIE aimed to identify and (ii) the corresponding radii associated with these berries. Figure 3.3 visually presents examples of the labeled data. A summarized representation of the manual measurement data associated with each bunch is presented in Table 3.1. This table includes (i) the unique bunch ID number, (ii) the total number of berries within the bunch (N_T), (iii) the count of visible berries in each image for the respective bunch (N_i, with $i = 1...3$, where $i = 1$ represents the image taken in standard configuration, $i = 2$ the image taken after rotating the bunch of 120°, and $i = 3$ the image taken after rotating it of 240°), (iv) the mean radius of berries (with their standard deviation) within the entire bunch in millimeters ($r_{\text{mean},T} \pm \sigma$), and (v) the average radius of the berries in pixels, computed manually based on the visible berries within each image ($r_{\text{mean},i}$).

(a) (b) (c)

Figure 3.3: (a-c) Image examples of the test dataset along with manual annotations overlaid.

Table 3.1: Summary of the validation data per bunch (ground truth), including (i) number of total berries (N_T), (ii) number of visible berries in each of the three images (0°, 120°, and 240° and N_1, N_2 and N_3, respectively), (iii) average radius of the bunch ($r_{mean,T}$, expressed in millimeters), and (iv) average radius of the visible berries in each image ($r_{mean,1}$, $r_{mean,2}$ and $r_{mean,2}$, all expressed in pixels).

Bunch ID	$\mathbf{N_T}$	$\mathbf{N_1}$	$\mathbf{N_2}$	$\mathbf{N_3}$	$\mathbf{r_{mean,T} \pm \sigma}$	$\mathbf{r_{mean,1}}$	$\mathbf{r_{mean,2}}$	$\mathbf{r_{mean,3}}$
Bunch_01	47	30	28	26	10.1 ± 0.35 mm	25 px	23 px	21 px
Bunch_02	52	37	32	41	9.3 ± 0.52 mm	21 px	21 px	19 px
Bunch_03	36	21	25	24	9.3 ± 0.44 mm	24 px	23 px	19 px
Bunch_04	39	23	23	21	9.2 ± 0.59 mm	25 px	22 px	19 px
Bunch_05	55	31	31	27	9.5 ± 0.57 mm	24 px	23 px	22 px
Bunch_06	51	32	35	29	10.1 ± 0.36 mm	23 px	22 px	25 px
Bunch_07	63	33	39	34	9.8 ± 0.42 mm	22 px	21 px	22 px
Bunch_08	52	30	34	29	10.1 ± 0.45 mm	22 px	21 px	21 px
Bunch_09	76	41	43	48	9.5 ± 0.40 mm	20 px	20 px	19 px
Bunch_10	40	29	24	24	9.5 ± 0.38 mm	22 px	21 px	20 px

Visible Berry Counting Validation

For the validation of the visible berry counting task, the model outputs obtained from the $K = 30$ test images must be compared with the real measured information summarized in Table 3.1. To achieve this, metrics such as the mean error (ME) and the mean percentage error (MPE) between the actual number of berries and the estimated number of berries within the test images were used (Equation (3.2)).

$$\begin{aligned} \mathrm{ME} &= \frac{1}{K}\sum_{k=1}^{K}\tilde{N}_k - N_k \\ \mathrm{MPE} &= \frac{1}{K}\sum_{k=1}^{K}\frac{\tilde{N}_k - N_k}{N_k} \end{aligned} \tag{3.2}$$

In Equation (3.2), $\tilde{N}_k$ and N_k represent the number of berries obtained from the algorithm and the real number of berries, respectively.

As further validation information, the berry counting output obtained on the test dataset should be compared with those acquired from the validation dataset (see Section 3.2.2). However, images in the WGISD dataset were captured in a field setting, often containing multiple clusters within a single image. As a result, the algorithm's estimation was applied to the image contents as a whole, rather than individual bunches. On the other hand, images in the test dataset were taken in a controlled set-up, each depicting a single cluster of a grape variety not present in the original training and validation datasets (see reference [1] for details about model training).

As a result, it is necessary to extrapolate the outcomes from the single-cluster-per-image scenario to the context of multiple clusters. To achieve this, we empirically verified that the prediction error of STEWIE on test images portraying a single bunch conforms to a normal distribution with a mean equal to ME and a standard deviation equal to σ_{E}. Thus, we designated the probability density function of the error for these single-bunch images as $f_{\mathrm{E}}(x) = \mathcal{N}(\mathrm{ME}, \sigma^2_{\mathrm{ME}})$. If

we analyzed an image with $B > 1$ bunches, the task would have involved computing the probability density function $f_{\mathrm{E}_B}(x)$ for errors in images containing B bunches. As a result, $f_{\mathrm{E}_B}(x)$ is the probability density function associated with the summation of B normal variables. Hence, the expression is reported in Equation (3.3) (the formula is valid under the fair assumption of independence between the errors for the bunches within the same image).

$$f_{\mathrm{E}_B}(x) = \mathcal{N}(B \cdot \mathrm{ME},\ B \cdot \sigma_{\mathrm{E}}^2) \tag{3.3}$$

Therefore, it is possible to estimate ME^B, MAE^B, and RMSE^B (root mean squared error) for an image containing B clusters by using Equation (3.4).

$$\begin{aligned} \mathrm{ME}^B &= \int_{-\infty}^{+\infty} x f_{\mathrm{E}_B}(x)\, dx \\ \mathrm{MAE}^B &= \int_{-\infty}^{+\infty} |x| f_{\mathrm{E}_B}(x)\, dx \\ \mathrm{RMSE}^B &= \sqrt{\int_{-\infty}^{+\infty} x^2 f_{\mathrm{E}_B}(x)\, dx} \end{aligned} \tag{3.4}$$

As the quantity of clusters within the validation images is not constant, we considered the average number $B = 14$ (this value was obtained by manually analyzing the images contained in the validation dataset).

Berry Radius Estimation Validation

To validate the model's capacity for accurate berry size estimation, the radius estimation results obtained on the test images are compared against the manually measured radius values (ground truth) listed in Table 3.1. The difference between the ground truth and the estimation constitutes the model's estimation error, computed using Equation (3.2). For this computation, the annotated mean radius $r_{\mathrm{mean},k}$ and STEWIE's estimation of the average radius $\tilde{r}_{\mathrm{mean},k}$ are used in place of N_k and $\tilde{N}_k$, respectively.

Volume Estimation Validation

As a final contribution, we derived the grape volume by using the estimated quantities of the number of berries and their average radius in pixels. This volume estimation can subsequently serve as a basis for farmers and wine producers to accurately calculate the yield [10, 11].

To validate the volume estimation, we need the effective volume of each bunch of the test dataset. The validation was performed considering (i) the volume of the entire bunch $V_{b,mm}$ and (ii) the volume of the visible part of the bunch $V_{I,px}$ (because 2D images depict only a portion of the overall berries in the bunch due to occlusions). The ground truth values for the volumes of the bunches in mm^3 were obtained by manually measuring the diameter of each grape berry of each bunch with a caliber (as described in Section 3.2.4), while the reference values for the computation of the volume of visible part of the bunch (in px^3) were obtained by manually annotating each grape berry on the images together with their diameters. Both volumes were derived considering the hypothesis of spherical berries.

Thus, we first computed the volume $\tilde{V}_{I,px}$ of the visible part of the bunch and compared it with the corresponding visible volume. From $\tilde{V}_{I,px}$, we then derived the volume of the entire bunch, $\tilde{V}_{b,mm}$.

To facilitate the subsequent discussion, we clarify the notation as follows:

- r_{mm} is the radius in metric coordinates of a berry in the bunch that was manually measured using a caliber;
- r_{px} is the radius in pixels of a berry present in an image that was manually measured from the image;
- $V_{I,px}$ represents the volume of the bunch b in px^3, considering only the berries visible in the image. This is approximated as $\sum_{n=1}^{N} \frac{4}{3}\pi r_{px,n}^3$;
- $V_{I,mm}$ represents the volume of the bunch b in mm^3, considering only the berries visible in the image. This is approximated as $\sum_{n=1}^{N} \frac{4}{3}\pi r_{mm,n}^3$, where $N < M$ represents the number of berries in the image;

- $V_{b,mm}$ defines the volume of the bunch b in mm^3. It is approximated as $\sum_{m=1}^{M} \frac{4}{3}\pi r_{mm,m}^3$. Here, M represents the total number of berries in the bunch, and $r_{mm,m}$ is the radius of the mth berry.

The estimated volume is computed using Equation (3.5), in which $\tilde{N}_k$ denotes the estimated number of berries and $\tilde{r}_{\text{mean}}$ is the mean radius estimated by STEWIE.

$$\tilde{V}_{I,px} = \tilde{N}_k \frac{4}{3}\pi \tilde{r}_{\text{mean}}^3 \tag{3.5}$$

To evaluate the accuracy of this estimation, we calculated ME and MPE (in pixels) between nominal volume $V_{I,px}$ and volume estimated from images $\tilde{V}_{I,px}$ (see Equation (3.2) for math computation). These metrics were derived by averaging the errors observed in individual images within the test dataset.

In a practical agricultural context, farmers are interested in obtaining a rough estimate of the total weight of their grape yield. To achieve this, the goal is to extend the estimated visible volume $\tilde{V}_{I,px}$ to estimate the volume of the entire grape cluster. Therefore, we need to determine $\tilde{V}_{b,mm}$, which represents the estimated volume of the entire grape cluster in metric units. To accomplish this, two steps are necessary: (i) convert $\tilde{V}_{I,px}$ to metric units $\tilde{V}_{I,mm}$ using the conversion factor C_{px-mm} that converts pixel units to millimeters (estimated by the camera calibration procedure as described in Section 3.2.3) and (ii) multiply the result by factor R, which takes into account the proportion of visible grapes with respect to the total grapes within cluster b in image k.

In this initial investigation, we proposed to calculate parameter R as the ratio between the volume of the entire grape bunch $V_{b,mm}$ and the volume of the visible grapes $V_{I,mm}$, averaged across all the images in our test dataset, which consists of 30 images. This parameter is crucial in our validation experiment because we measured the entire bunch volume manually; however, only a portion can be seen in the images since some berries were be obscured by the foreground ones, thus leading to an underestimation of the visible volume in the images. Hence, parameter R was used as a correction factor. It is worth noting that we introduced

some bias into the calculation by utilizing the ground truth information from the dataset. Ideally, we would select a small subset of grape bunches to calculate the conversion factor R and then validate its applicability on the remaining dataset. However, due to the constraints of our limited dataset, we chose the approach described above, which remains a reasonable and practical solution for this preliminary analysis. As a result, the estimated volume of the whole grape bunch in metric units is computed by using Equation (3.6).

$$\tilde{V}_{b,mm} = \tilde{V}_{I,px} \cdot C_{px-mm}^{3} \cdot R \tag{3.6}$$

This information serves as the final output for farmers, aiding them in estimating the total yield of their grape harvest. It is important to note that this value is subject to uncertainty.

3.2.5 Uncertainty Evaluation for Volume Estimation

Since the final output for the farmers is $\tilde{V}_{b,mm}$, it is necessary to evaluate the uncertainty of each variable that contributes to its calculus by using the "Guide to the Expression of Uncertainty" (GUM, linear propagation, simplified approach) [12, 13].

Before estimating the uncertainty of $\tilde{V}_{b,mm}$, we first need to estimate the uncertainty of $V_{I,px}$ (obtained from Equation (3.7)) and $V_{b,mm}$ (obtained from Equation (3.8)) to ensure that they can be taken as reference values. To achieve this, we need to evaluate the uncertainty of (i) r_{px}, and (ii) r_{mm}. For the manually annotated radius (r_{px}) and the caliber measured radius (r_{mm}) uncertainties, we considered, respectively, $\frac{a}{\sqrt{3}}$, with a being 1 px (image resolution) and 0.3 mm (calculated through repeated measures on the same berries). The uncertainties of $V_{I,px}$ and $V_{b,mm}$ were computed by applying GUM to Equations (3.7) and (3.8).

$$V_{I,px} = \sum_{k=1}^{N_i} \frac{4}{3} \cdot \pi r_{px}^{3} \tag{3.7}$$

$$V_{b,mm} = \sum_{k=1}^{N_T} \frac{4}{3} \cdot \pi r_{mm}^3 \tag{3.8}$$

To compute the uncertainty of $\tilde{V}_{b,mm}$, we need to consider following variables: (i) conversion factor C_{px-mm} (and thus the camera–grape distance d and focal length f), (ii) the average ratio R, and (iii) the estimated grape bunch volume $\tilde{V}_{I,px}$ expressed in px^3.

We chose to associate to d uncertainty $\sigma_d = 25$ mm. This assumption was defined because the positioning of the grape bunch with respect to the camera varies depending on where it is located on the vine (the vine distance was fixed to 500 mm from the camera). We empirically defined σ_d as the half thickness of the vine considered. In real-case applications, this parameter could be set to higher values to account also for the uncertainty on d. The focal length f (in pixels) was estimated through a standard camera calibration procedure as described in Section 3.2.3 with uncertainty equal to $\sigma_f = 2$ px.

The uncertainty of parameter R was computed as the standard deviation of the observed ratio between the volume of the entire grape bunch $V_{b,mm}$ and the volume of the visible grapes $V_{I,mm}$, averaged across all the images in the test dataset. This resulted in a value of $\sigma_R = 0.32$.

The uncertainty of $\tilde{V}_{I,px}$ (Equation (3.5)) was obtained by computing the RMSE between the estimated visible pixel volumes for each Image I in the test dataset, $\tilde{V}_{I,px}$, and their corresponding reference volume $V_{I,px}$ (Equation (3.7)). We recall that the main difference between the two volumes stands in the radius used in the formulation, which is the predicted mean berry radius obtained from STEWIE in the case of $\tilde{V}_{I,px}$ and the manually annotated radius of each visible berry in the case of $V_{I,px}$.

To compute the uncertainty of the total estimated volume of each bunch b ($\tilde{V}_{b,mm}$), we applied GUM to Equation (3.6).

3.3 Results and Discussion

3.3.1 Model Validation

Model performances in berry counting, radius, and volume estimation are shown in Figure 3.4. For the berry counting task applied to the test dataset (red box plot in Figure 3.4), the resulting mean error ME is -1.57 with a standard deviation σ_{E} equal to 1.9. The mean percentage error MPE is equal to -5% with a standard deviation of 6.3%. For the sake of completeness, we also present the individual count for each image in Table 3.2. The negative value highlights that STEWIE tends to underestimate the number of berries, probably due to occlusions (e.g., background berries completely or partially hidden by foreground berries).

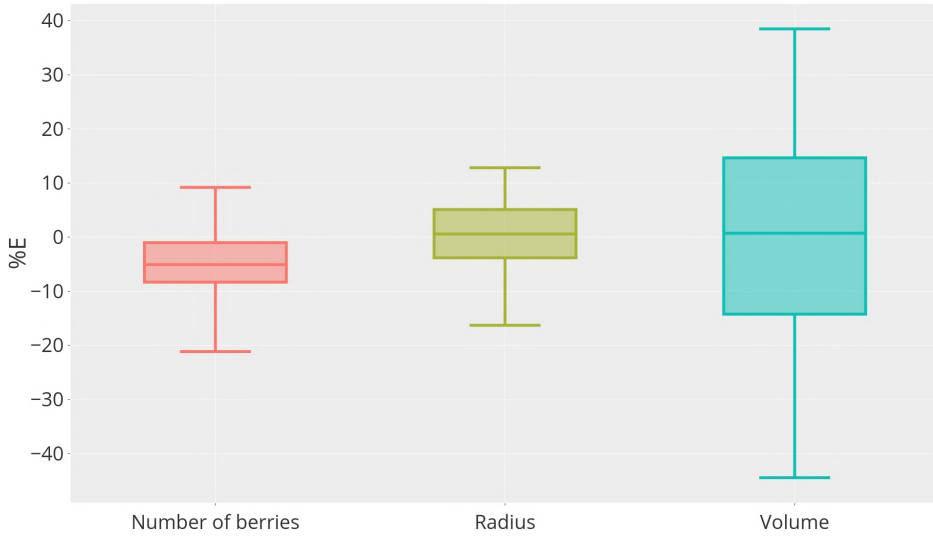


Figure 3.4: Boxplot of the relative error RE_k in % computed for the number of visible berries, the value of the average radius, and the visible volume of the bunch depicted in the test images.

Table 3.2: Estimated number of berries for each of the three images of the 10 bunches comprising the test dataset. Inside the parentheses, the difference with respect to the ground truth (refer to Table 3.1 for comparison).

Bunch ID	$\tilde{\mathbf{N}}_\mathbf{1}$	$\tilde{\mathbf{N}}_\mathbf{2}$	$\tilde{\mathbf{N}}_\mathbf{3}$
Bunch_01	30 (0)	28 (0)	26 (0)
Bunch_02	32 (−5)	35 (+3)	38 (−3)
Bunch_03	20 (−1)	24 (−1)	23 (−1)
Bunch_04	22 (−1)	23 (0)	20 (−1)
Bunch_05	28 (−3)	28 (−3)	21 (−6)
Bunch_06	33 (+1)	37 (+2)	36 (−3)
Bunch_07	31 (−2)	36 (−3)	31 (−3)
Bunch_08	28 (−2)	30 (+1)	29 (0)
Bunch_09	38 (−3)	42 (−1)	45 (−3)
Bunch_10	24 (−5)	23 (−1)	24 (0)

Table 3.3 shows the comparison between the validation and test datasets following the procedure in Section 3.2.4 to obtain comparable results between single-cluster-per-image and multiple-clusters-per-image scenarios. To enhance clarity and emphasize distinctions, the results are presented by dividing the outcomes associated with the validation dataset based on the grape variety. Results obtained using the validation dataset are fully comparable with those obtained with the test dataset. The MAE column indicates that counting errors may reach around 32 berries for an image containing an average of 14 clusters. To understand whether this error is acceptable, it is essential to determine the total number of berries in the validation and test datasets. Based on the data shown in Table 3.1, we hypothesize that the average number of berries for each cluster is 50 (obtained averaging column N_T of Table 3.1). Expanding this value to 14 clusters, we obtain an average number of berries equal to 700. This value is utilized to compute the normalized MAE (column MAE_{norm} in Table 3.3). The average MAE is equal to 3.1%, with a standard deviation of 0.8%. This number is entirely suitable for the application considered.

Table 3.3: Evaluation metrics of the berry counting task computed for both test and validation images, divided by grape variety. I_T refers to the number of images of the variety in the dataset. The values corresponding to the test dataset are the estimated values derived from the analysis detailed in Section 3.2.4. Column MAE_{norm} is computed considering an average number of berries in each image equal to 700.

Variety	**Dataset Used**	**ME**	**MAE**	$\mathbf{MAE_{norm}}$	**RMSE**	$\mathbf{I_T}$
Chardonnay	Validation	-26.8	32.2	4.5%	45.5	13
Cabernet Franc	Validation	-0.7	17.0	2.4%	20.5	22
Cabernet Sauvignon	Validation	4.0	21.1	2.9%	30.7	14
Sauvignon Blanc	Validation	5.7	23.8	3.3%	31.6	15
Syrah	Validation	-8.6	16.5	2.3%	21.7	11
Flame	Test	-21.8	21.9	3.1%	23.1	30

Regarding the mean radius estimation task (green box plot in Figure 3.4), STEWIE achieves a mean estimation error ME of 0.15 px with a standard deviation of 1.5 px, corresponding to a mean percentage error MPE of 0.8% with a standard deviation of 7%, which is a promising result.

Regarding the volume estimation, we first compare the reference visible volume using the manual annotations, $V_{I,px}$, with the estimated visible volume obtained using the mean berry radius predicted by STEWIE, $\tilde{V}_{I,px}$ (blue box plot in Figure 3.4). We obtain a mean estimation error ME of $-20.7 \cdot 10^3$ px^3 with a standard deviation of $2.9 \cdot 10^5$ px^3 and a mean percentage error MPE of 0.36% with a standard deviation of 20.8%. Even if the average error is almost null, results on volume estimation (in pixels) show a high variability. This inconsistency in results suggests that the low bias of the volume estimates may not be reliable.

As described in Section 3.2.4, once the *visible* volume estimation $\tilde{V}_{I,px}$ is

evaluated, we examine the volume of whole bunches $V_{b,mm}$. Figure 3.5 shows the estimation error E_k obtained as the difference between estimated volume $\tilde{V}_{b,mm}$ and nominal volume $V_{b,mm}$ for each image k in the test dataset, coupled with the corresponding uncertainty. Uncertainties of each quantity considered are computed as described in Section 3.2.5. Figure 3.5 shows that the measured volume of almost every grape bunch is compatible with its nominal value.

For each grape bunch, we compute the coefficients of variability (CoV) (i.e., the ratio between the standard deviation and the mean of each measure). All measurements show high variability: average CoV within 30 measurements equals 24% with a standard deviation of 7%. This effect is partially due to the variability of $\tilde{V}_{I,px}$. Other parameters that affect this result are the uncertainty of (i) C_{px-mm} and (ii) R. To understand this result, a further analysis of the uncertainty associated with the measurement must be conducted, as described in the following section.

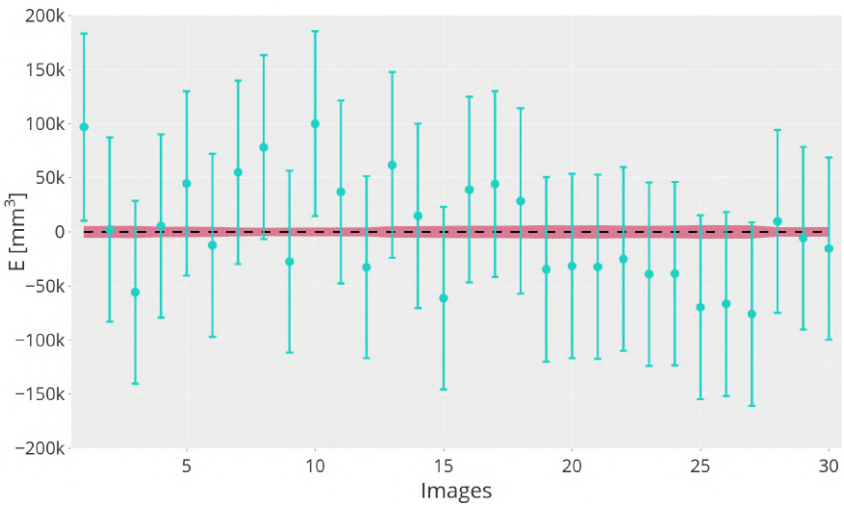


Figure 3.5: Total bunch volume error E_k (difference between estimated volume $\tilde{V}_{b,mm}$ and nominal volume $V_{b,mm}$) for each image k in the test dataset, coupled with the corresponding uncertainty. The shaded red area corresponds to the 95% confidence interval of the ground truth.

3.3.2 Uncertainty Analysis

As described in Section 3.2.5, we computed the measurement uncertainty of every parameter that plays a role in the overall computation of the volume's uncertainty. To compute the uncertainty on the total estimated volume of each bunch b ($\tilde{V}_{b,mm}$), we applied GUM to Equation (3.6).

Table 3.4 shows the summary of the variables coupled with their corresponding uncertainty σ and a brief description of the uncertainty source. The last column shows their average percentage contribution to the overall uncertainty (UPC). From this analysis, it is evident that the most impact on the overall uncertainty was due to $\tilde{V}_{I,px}$, which is estimated considering the mean berry radius of the visible berries produced by STEWIE.

Variables	Uncertainty	Definition and Reason of Uncertainty	UPC
d	25 mm	Uncertainty set considering the vine thickness and the eventual cluster misplacement that could modify the default value of d.	22.1%
f	2 px	Uncertainty depends on the quality of the images and of the pattern used for the camera calibration procedure.	0.3%
R	0.32	Uncertainty set as the standard deviation of the values that were averaged to compute R (e.g. the ratios between the visible and the total volume of the bunches for each photo).	32.6%
$\tilde{V}_{I,px}$	$2.8 \cdot 10^5$ px^3	Uncertainty set as the RMSE between the estimated pixel volumes for each Image I in the test dataset, $\tilde{V}_{I,px}$, and their corresponding reference volume $V_{I,px}$.	45%

Table 3.4: Sources of uncertainty definition. Each variable is shown with its corresponding uncertainty and UPC for quick reference.

The uncertainty on the estimated volume in pixel resulted in $\sigma_{\tilde{V}_{I,px}} = 2.8 \cdot 10^5$ px^3 and the average uncertainty on the reference volume in pixel results in $\sigma_{V_{I,px}} = 1.8 \cdot 10^4$ px^3. We adopted the average uncertainty on the reference because, by applying GUM, we obtained individual uncertainties for each $\mathrm{V}_{k,px}$. It is worth noting that the uncertainty on the estimated visible volume, $\sigma_{\tilde{V}_{I,px}}$, exceeds by more than an order of magnitude the average uncertainty on its ref-

erence, $\sigma_{V_{I,px}}$.

While the uncertainty on the visible volume ($\sigma_{\tilde{V}_{I,px}}$) was the most prominent (45%), it can be noted that the uncertainties on the camera–grape cluster distance d (σ_d) and on the total/visible volume ratio R (σ_R) had a combined impact that was greater than 50%. By adopting strategies to increase the confidence level on d and R, it could be possible to halve the uncertainty on the visible volume estimation formulation.

For each of the $K = 30$ images of the test dataset (depicting single grape bunches b), it is shown in Figure 3.5 that the uncertainty on the estimated total volume of the bunch $\tilde{V}_{b,mm}$ was greater than the reference total volume $V_{b,mm}$ by more than an order of magnitude (considering all bunches, we obtained a mean uncertainty of 42 mm^3, approximately equal to the 20% of the estimated volume). This result is certainly not satisfactory and could be improved by reducing the uncertainty related to the estimation of the total/visible volume ratio R and of the camera–grape cluster distance d. To this aim, a possible solution is to adopt depth cameras in a 2D–3D fusion fashion to always know the actual position of the bunch with respect to the camera (d) and consequently design a better formulation for parameter R. Moreover, the volume was elevated at the power of three, which greatly emphasizes the effect of small errors in the estimation of the average berry radius. Additionally, it is worth mentioning that the RMSE on the volume estimation appeared unbiased (average error close to zero). Thus, averaging multiple bunches could lead to favorable outcomes in whole-orchard analysis.

3.4 Conclusions

In this article, we conducted a metrological validation of the performance of the weakly supervised neural network named STEWIE introduced in our previous work [1], which directly outputs both the number of individual grape berries and their average radius from 2D images. This is a novel feature not yet explored by other works, especially for small fruits such as grape berries. The contribution of

this article stands in the thorough validation and uncertainty evaluation of the model's performance, a topic often overlooked in precision agriculture research.

The validation was conducted on the two outputs of the model: (i) the visible berry counting in the images and (ii) the corresponding berry radius estimation. From these two parameters, it is possible to compute the overall grape bunch volume, which is the key information needed by vinegrowers to accurately estimate the yield. To assess which parameter contributes to the most uncertainty in the final volume estimation, we applied the GUM and derived the UPC of each parameter. This analysis highlighted that half of the total uncertainty on the volume is due to the camera–object distance d and parameter R used to take into account the proportion of visible grapes with respect to the total grapes in the grape cluster. As a result, by using more reliable sensors to measure d such as depth cameras, our model performance improves.

Finally, since winegrowers are more interested in the whole orchard yield volume information while taking into account the uncertainty of the measurement at the same time, we aim to further improve STEWIE model and incorporate the uncertainty estimation on the final volume output in its design. The complete system will be developed and deployed on a robust embedded device able to acquire every information needed coupled with the corresponding frames so that the whole orchard can be analyzed effortlessly.

Bibliography

[1] Davide Botturi, Alessandro Gnutti, Cristina Nuzzi, Bernardo Lanza, and Simone Pasinetti. STEWIE: eSTimating grapE Berries Number and Radius From Images Using a Weakly supervIsed nEural Network. In *2023 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pages 277–282, 2023.

[2] Lufeng Luo, Wentao Liu, Qinghua Lu, Jinhai Wang, Weichang Wen, De Yan, and Yunchao Tang. Grape berry detection and size measurement based on edge image processing and geometric morphology. *Machines*, 9(10):233, Oct 2021.

[3] Laura Zabawa, Anna Kicherer, Lasse Klingbeil, Reinhard Töpfer, Heiner Kuhlmann, and Ribana Roscher. Counting of grapevine berries in images via semantic segmentation using convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 164:73–83, 2020.

[4] Santos Thiago, Leonardo de Souza, Andreza dos Santos, and Sandra Avila. Embrapa Wine Grape Instance Segmentation Dataset – Embrapa WGISD, 2019. Public dataset on Zenodo.

[5] Geng Deng, Tianyu Geng, Chengxin He, Xinao Wang, Bangjun He, and Lei Duan. TSGYE: Two-stage grape yield estimation. In Haiqin Yang, Kitsuchart Pasupa, Andrew Chi-Sing Leung, James T. Kwok, Jonathan H. Chan, and Irwin King, editors, *Neural Information Processing*, pages 580–588. Springer International Publishing, Cham, 2020.

[6] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[7] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1106–1112, 1997.

[8] The Mathworks, Inc., Natick, Massachusetts. *MATLAB version 9.3.0.713579 (R2017b)*, 2017.

[9] Jean-Yves Bouguet. Camera calibration toolbox for matlab. 2022.

[10] Brittany Komm and Michelle Moyer. *Vineyard yield estimation.* Washington State University Extension, 2015.

[11] André Barriguinha, Miguel de Castro Neto, and Artur Gil. Vineyard yield estimation, prediction, and forecasting: A systematic literature review. *Agronomy*, 11(9):1789, Sep 2021.

[12] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML. Evaluation of measurement data — Guide to the expression of uncertainty in measurement. Joint Committee for Guides in Metrology, JCGM 100:2008.

[13] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML. Guide to the expression of uncertainty in measurement — Part 6: Developing and using measurement models. Joint Committee for Guides in Metrology, JCGM GUM-6:2020.

Chapter 4

Depth from Monocular RGB Cameras

4.1 From Localization to Depth: Enhancing RGB-Based Measurements in Agriculture

In precision agriculture, localization is fundamental for contextualizing sensor data and generating georeferenced maps, crucial for monitoring parameters such as grape and berry clusters, wood volume, and bud distribution. Accurate localization improves agronomic decision-making by ensuring precise tracking of these elements.

To achieve this, various localization techniques are employed:

- **T265 Tracking Camera**: Provides real-time odometry and motion tracking.
- **Aruco Markers**: Serve as ground-truth references for calibration and validation.
- **IMU-Integrated Sensors**: Compensate for motion noise and vibrations.

- **RTK GNSS**: Enables high-precision spatial referencing in vineyards.

Although these methods establish the spatial position of the camera, they do not provide depth information, limiting the ability to fully characterize objects in the scene. This is particularly relevant when using standard RGB cameras, where localization alone cannot yield accurate three-dimensional representations. Depth estimation is essential for translating pixel-based measurements into real-world dimensions, ensuring reliable evaluations of size, area, and volume.

Integrating depth information enhances agricultural applications by refining RGB-based measurements. For instance, accurate depth data improves the estimation of grape cluster dimensions and bud positions, while enabling the generation of point clouds for detailed 3D reconstructions. These capabilities support improved vineyard management through more precise and actionable data.

A viable approach to depth estimation from monocular RGB imagery involves analyzing object motion in the image plane. When a camera moves, objects at different distances exhibit varying optical flow displacements. By leveraging this principle and incorporating known camera motion, depth can be inferred without dedicated depth sensors.

In structured agricultural environments—such as vineyards where sensors are mounted on moving vehicles—this method offers a cost-effective way to enhance depth perception. Estimated depth, combined with RGB imagery, enables:

- More accurate grape cluster and berry size estimations.
- Improved segmentation and tracking of buds and wood structures.
- Expanded usability of RGB cameras for agricultural monitoring.

This study explores the integration of optical flow for depth estimation in agricultural scenarios, evaluating its reliability and uncertainty. By doing so, it aims to provide a practical and cost-effective alternative for real-world applications where low-cost cameras are deployed in the field.

4.2 Introduction

Outdoor depth estimation can be achieved in several ways, depending on the hardware and software adopted. A popular and low-cost choice is the adoption of depth cameras leveraging stereo vision. However, the quality of the output (both the color and depth images) is often insufficient, especially when employed in applications where the camera moves fast, introducing defects due to the shutter and acquisition speed. In the case of high-speed movements, a good color camera paired with high-end optics is necessary to cope with the speed of the moving scenario to be acquired. Starting from good-quality images, the depth estimation problem is addressed by computer vision (CV) techniques. One of the most popular methods is Optical Flow (OF), first developed in 1981 [1, 2]. OF is defined as the apparent motion of objects in a sequence of images caused by the relative motion between the scene captured in them and the camera. Therefore, the problem encompasses several variables such as ambient illumination, objects' texture, and difficult geometrical shapes for which occlusions may happen, producing incorrect OF estimates [3, 4, 5]. In general, OF is obtained as a map of vectors indicating, for each pixel in the image, the corresponding apparent motion and its intensity. OF is used to estimate the speed and depth of both slow and fast phenomena and is applied in several fields, such as healthcare [6], robotics and moving vehicles in general [7], industry [8], and agriculture [9, 10]. Modern approaches for solving OF issues exploit Deep Learning (DL) to improve the estimates, especially in the case of depth estimation [11, 12]. Other recent advancements in the field of monocular depth estimation are presented in [13, 14, 15, 16]. These works extensively adopt DL models to estimate depth from monocular images without employing OF and are considered the state-of-the-art by the CV community. However, their depth output is not stable and it is not based on a measurement, not to mention the computational requirements needed to run those models on mobile and embedded devices. Some of those models do not produce outputs in metric coordinates either, making it difficult for untrained personnel to use those models in real in-field applications for which near real-time computation is required. Generally, it is still tricky to couple depth estimates with measurement uncertainty, typically treated as a confidence measure

[17] that does not adhere to the "Guide to the Expression of Uncertainty" manual [18, 19]. Moreover, monocular depth estimation research is mostly rooted in the CV community, which focuses more on the software and mathematical aspects of the problem and less on practical needs for the untrained end-user to obtain a reliable depth measure, such as which hardware to choose according to specific characteristics and what are the limitations of the measurement set-up to keep in mind.

Focusing on the agricultural sector, however, the general problem of OF estimation can be simplified since the measurement environment is more constrained. Farming vehicles, such as tractors and fruit-picking robots, move linearly along the rows of the field at a constant speed, which is not higher than 5 km/h in the case of big and heavy tractors and close to 2 m/s for agricultural robots. In addition, modern tractors are being designed to be autonomously driven, requiring accurate tractor-row distance estimation to adapt to the specific field they work on [20]. Hence, by mounting a camera on the moving vehicle (for which the speed is known thanks to the vehicle's encoder), it is possible to acquire a sequence of images for which the temporal and spatial relationships are known as well, thanks to timestamps and GNSS sensors that geo-localize each frame. In addition, by mounting the camera orthogonal to the vehicle movement direction so that it looks at the canopy, its reference frame primarily moves along its Y-axis. By focusing solely on the Y-Z plane with the motion vector confined within it, these assumptions allow for a simplified model implementation.

This research aims to demonstrate with a practical laboratory experiment that mimics the agricultural scenario how OF performs at different camera speeds and relative camera-object depths. To this end, a simplified version of OF was adopted, inspired by the Structure from Motion (SfM) CV problem, which allows the reconstruction of complex 3D structures from 2D multi-view images of the scene collected by a moving camera [21]. Traditional SfM techniques are computationally intensive and hardware-demanding, typically adopted for applications such as land reconstruction, architecture, and construction. Therefore, in this paper, a custom model that is easier to understand for end-users was designed. In particular, the focus of the article stands on the models' validation

and uncertainty analysis, which ultimately provides ready-to-use information for the end-user (e.g., the farmer) to both choose the right camera for the target application and design the measurement set-up and constraints. Finally, the code used for this work is made publicly available on GitHub at [22].

4.3 Materials

4.3.1 Equipment and experimental set-up

A scheme of the acquisition set-up used for this work is shown in Figure 4.1, depicting a robot, a color camera, and a custom-made target. Images taken from the acquisition set-up can be found in Figure 4.2.

The moving vehicle is simulated by a robot (Universal Robots UR10e) mounted upside-down in the workspace described in [23]. On the robot's end-effector a color camera (GoPro Hero 11 Black) was mounted, tasked with the acquisition of color images during movement. The robot was programmed to move horizontally at 5 constant speeds ($V_1 = 0.25$ m/s, $V_2 = 0.5$ m/s, $V_3 = 0.75$ m/s, $V_4 = 0.94$ m/s, $V_5 = 0.97$ m/s) with a horizontal travel distance of 1.45 m. The speeds tested were selected according to the common operating speed of farming vehicles.

The camera has a 1/1.9" CMOS sensor with resolution of 5599×4927 px, and F2.5 lens aperture. To increase the camera frame rate, during acquisition the camera was set to a resolution of 2704×1520 px with an aspect ratio of $4:3$, allowing it to acquire frames at 60 frames per second (fps). The lens distortion coefficients and the internal camera matrix (containing the position of the image center C_X and C_Y [px] as well as the focal length dimension in both directions f_X and f_Y [px]) were computed by performing a calibration procedure using a chessboard pattern [24]. It is worth noting that the pixels of the chosen camera have a square shape, hence $f_X = f_Y = f$.

The custom-made target measurand is made of an aluminum bar (metallic support) placed at a height of $H = 1.3$ m from the ground, on which 5 smaller

bars of different sizes (B_1 = 10 cm, B_2 = 20 cm, B_3 = 40 cm, B_4 = 55 cm, B_5 = 80 cm) were fixed orthogonally. The target was crafted to simulate agricultural rows where plants grow at different depths than the guiding canopy. The metallic support was positioned at 4 different locations from the robot reference frame (RF), D_1 = 115 cm, D_2 = 135 cm, D_3 = 155 cm, D_4 = 165 cm. These distances were chosen to simulate different working conditions (e.g., vehicles moving at different distances from the plants growing in the row, and different row distances).

On top of each bar, an ArUco marker [25, 26, 27] of size 45 × 45 cm was positioned to be orthogonal to the camera during acquisition (M_1, M_2, M_3, M_4, and M_5). ArUco markers are square matrixes of black and white cells that easily represent a location and an orientation at the same time according to the deformation of the matrix pattern. Each ArUco marker is unique and procedurally generated by the related library, which includes algorithms and functions to retrieve the information of a specific marker from an image. In this work, the ArUco markers serve as the depth ground-truth. The ArUco OpenCV library was employed to extract the markers' depth [25, 26, 27]. A picture of the markers used can be seen in Figure 4.2.

4.3.2 Data acquisition

A total of 20 tests were conducted (4 D target positions × 5 V camera velocity). Each camera acquisition is composed of a positive and a negative camera movement along the Y axis. Instead of saving each acquired frame independently, a more computationally efficient solution was defined. A single video in MP4 format was recorded, each video including frames related to a specific combination of D and V, for both movements along $+Y$ axis and $-Y$ axis. The customized algorithm detailed in Section 4.4.2 extracts individual frames from the video, splits movements along $+Y$ axis and $-Y$ axis, and computes the distance from the camera (depth Z m) of each ArUco marker. Subsequently, the displacement along the Y-axis (ΔY px) of the marker's center is calculated between two consecutive frames.

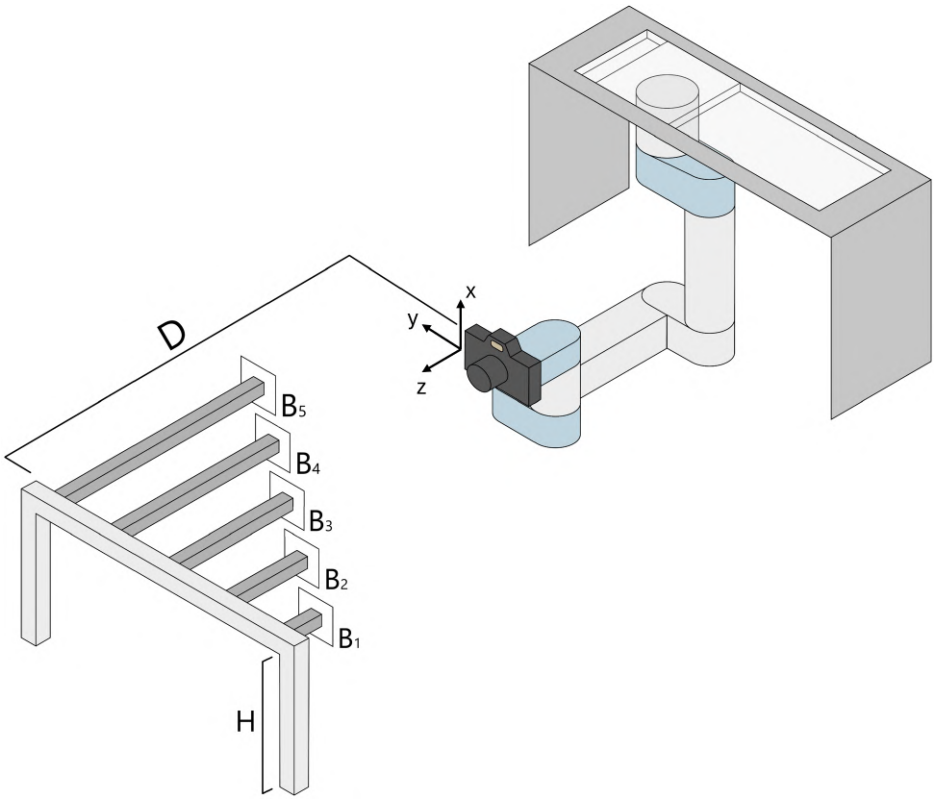


Figure 4.1: Graphical scheme of the set-up adopted in this work, highlighting relevant parameters and sizes of the involved equipment.

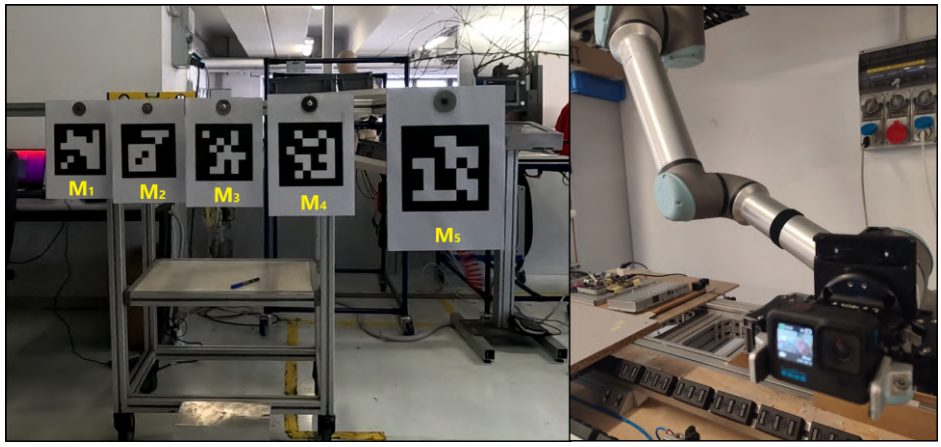


Figure 4.2: Example of the set-up and the equipment adopted in this work. ArUco markers are highlighted in yellow.

The number of frames in a single trial varies depending on the camera's velocity, as the robot follows a fixed path, with an average of 300 frames. Overall, a total of $60,000$ data points were collected (5 robot speeds $\times$ 2 robot movement directions $\times$ 4 relative distances of the metallic support from the camera $\times$ 5 ArUco markers at different depths $\times$ 300 frames).

4.4 Methods

4.4.1 Model definition

Due to the relative motion between a point in the world's reference frame (WRF) and the camera's reference frame (CRF), a displacement δy is observed in the image plane between two consecutive frames. This displacement is corrected for lens distortion through a camera calibration process (such as [24] or similar approaches). The model depicted in Figure 4.3 illustrates the projection of a point of interest onto the camera plane.

On the Y-axis displacement, there are two similar triangles $(f, \delta y)$ and $(Z, \Delta Y)$.

Based on their similarity properties, the following equation is obtained:

$$\frac{\Delta Y}{\delta y} = \frac{Z}{f_Y} \tag{4.1}$$

where ΔY is the displacement of the point in the CRF [m], δy is the displacement of the point in the image plane [px], Z is the depth corresponding to the point [m], and f_Y is the focal length of the camera [px].

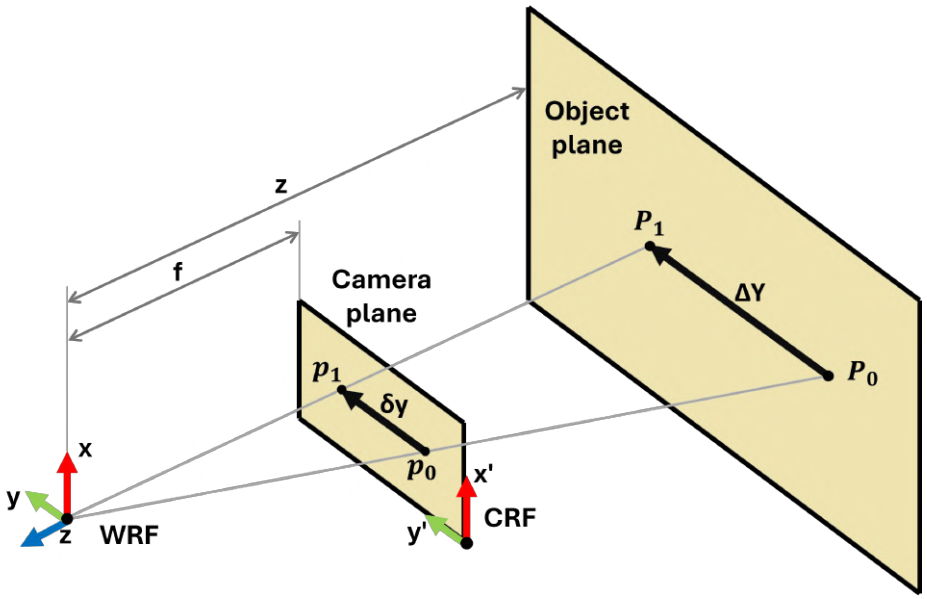


Figure 4.3: Scheme depicting the projection of a point from the real world to the image plane assuming a pinhole camera model.

By applying Eq. 4.1, it is possible to estimate the point's depth (Z) based on its pixel displacement in the image plane δy, given its relative displacement with respect to the CRF ΔY. To accurately calculate δy with respect to a specific point or object within an image, several challenges arise in detection and tracking. Detection challenges include reliably identifying specific points or objects despite varying lighting conditions and similar-looking objects. Ensuring consistency in detecting the same points across multiple frames is crucial for

accurate displacement calculation.

In tracking, the main issue is maintaining the continuity of the detected object across successive frames, which requires robust algorithms capable of handling rapid movements, changes in scale, and rotation. Additionally, tracking algorithms may experience drift over time, leading to deviations from the actual position due to cumulative errors.

Advanced techniques are typically employed to address these challenges. Feature extraction methods provide robust features invariant to changes in scale, rotation, and illumination, aiding in detection and tracking. Optical flow (OF) algorithms calculate the apparent motion of objects between consecutive frames, directly providing δy by analyzing pixel displacement. Marker-based tracking, using fiducial markers, offers a reliable method as these markers are easily detectable and their positions can be accurately determined. Integrating these methods improves the accuracy and reliability of δy calculations. This displacement can be related to the original speed of the object tracked according to the formula $S = V \cdot T$, where S represents the space traversed by the tracked object, V refers to the movement velocity, and $T = t_2 - t_1$ refers to the time between two consecutive frames. Two versions of this relationship can be defined accordingly: $\Delta Y = V_{camera} \cdot T$ [m/s] and $\delta Y = V_{image} \cdot T$ [px/s]. Therefore, the speed of the object in the WRF (V_{camera}) is related to the apparent speed of the object in the CRF (V_{image}).

By substituting these relations into Eq. 4.1 and simplifying for T (purposely considered equal for the two terms), Z can be obtained as:

$$Z = \frac{V_{camera} \cdot f_Y}{V_{image}} \tag{4.2}$$

This equation will be referred to as the *Analytical model* definition.

In this work, the aim is to simplify the relationship for Z estimation using only the OF output, treating V_{camera} as a constant rather than an unknown variable. By assuming a constant V_{camera} during movement, the idea is to obtain a general parameter K that allows end users to estimate Z with sufficient confidence for

the target application using the data output of OF to obtain V_{image}. Following this, the *Experimental model* definition is as follows:

$$Z = \frac{K}{V_{image}} \tag{4.3}$$

where parameter K is unknown [m · px/s].

4.4.2 Depth computation

The 20 videos v of the tests (see Section 4.3.2) were processed leveraging a variety of open-source functions available on the OpenCV library. Processing code was developed in Python and is publicly available at the link provided in reference [22]. Each video is analyzed independently, and the frames contained in it are not saved on disk; instead, to save space, a data extraction procedure was applied to analyze the contents of the frames and save the outputs in separate files in CSV format.

Each video sequence v contains a set of frames i acquired at time t_i. Each frame is processed individually following a two-block operational procedure illustrated in Figure 4.4 (first block in blue, second block in pink). The first block involves calculating the pose of the markers thanks to a set of image processing analyses (yellow block inside the blue block of Figure 4.4), followed by the second block, which focuses on computing the optical flow for each marker. The first block structure is the following:

1. *Conversion to grayscale.* The current frame I_i is converted from color to grayscale to facilitate the subsequent operations.

2. *Brightness and contrast correction.* The frame's contrast α and brightness β are optimized to improve the image's quality. These parameters are defined to ensure that the markers can be detected with sufficient accuracy by the ArUco library functions. The best values were experimentally found and are equal to $\alpha = 2$ and $\beta = 5$ according to the general illumination of the working conditions of the experimental setup. To read further about how

these two parameters are defined and used in OpenCV, please refer to the official documentation in [28].

3. *ArUco markers finding.* The five markers m in the image are identified using a specific set of functions in the OpenCV ArUco library [25, 26, 27].

4. *Computation of apparent depth.* Using the library, the depth $Z_{i,m}$ of each marker is calculated, providing an estimate of the marker's Z coordinate relative to the camera's reference system.

5. *Computation of markers' centers.* For each detected marker m, its center coordinates on the image plane, $C_{i,m} = (x'_{i,m}, y'_{i,m})$, are calculated. Coordinates x' and y' are different from the 3D coordinates (X, Y, Z) that involve marker and camera calibration since they are solely related to the marker's detection on the image plane.

Outputs of the first operational block are the ground truth coordinates of the ArUco markers acquired, $(x'_{i,m}, y'_{i,m}, Z_{i,m})$.

The second operational block has been defined to compute pixel displacements between consecutive frames. To do so, starting from the intermediate output table, the OF algorithm is applied on pairs of images I_{i-1} and I_i (thus beginning the counter from image $i = 2$), using as tracked markers the centers of the markers $C_{i,m}$ detected before. This produces another value called $OF_{i,m}$ representing the pixel displacement δY between consecutive images.

For each video $v = 1...20$ the overall procedure produces a table Tab_v of $N-1$ rows $\times$ 11 columns (time instant t_i plus the 5 Z-coordinates of the markers' centers $Z_{i,m}$ and the 5 OF each marker's center $OF_{i,m}$). Each Tab_v, containing the OF data stream over time, is saved in a CSV file.

4.4.3 Signals synchronization and filtering

Considering the analytical model definition described by Eq. 4.2, in the experiments V_{camera} was equal to the robot's speed. However, even if the robot is

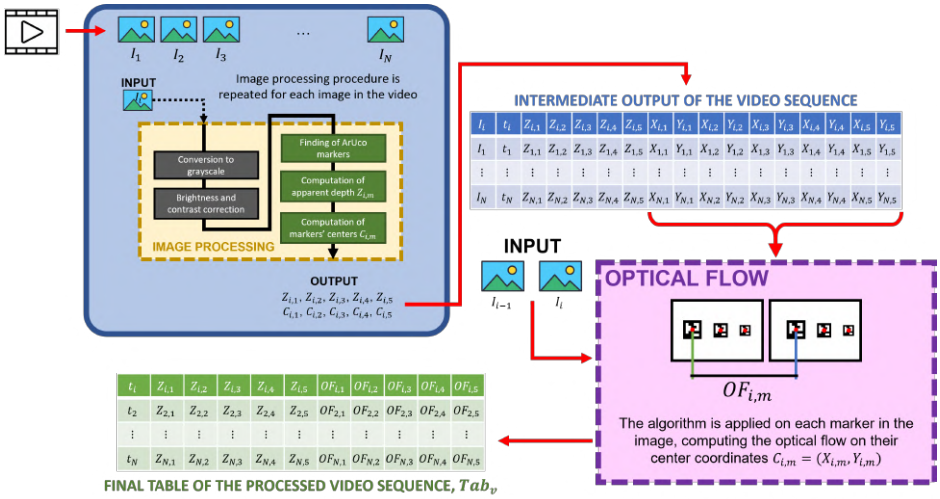


Figure 4.4: Graphical scheme of the processing procedure divided into two operational blocks (blue, pink). The image processing step (yellow) is repeated for each image $i = 1...N$ in the video sequence v, producing an intermediate output table. Then, the OF algorithm is applied to image couples, starting from image $i = 2$ and tracking changes of the markers' centers (obtained from the first operational block in blue).

theoretically actuated at a constant speed, some portions of its movement include acceleration and deceleration (at the start, when changing direction, and at the end of the movement). Ideally, the robot's movement during the experiment and the camera acquisition should be synchronized. However, the robot signal was obtained from its encoders, which produce denser data compared to the data obtained from the image-based analysis of Section 4.4.2; thus, the two data streams (the ArUco markers' centers positions in the image plane in Tab_v and the robot position R_v over time) are not synchronized and contain a different number of points. To address this issue and obtain synchronized data streams, the following procedure was applied:

1. The robot's signal obtained from the encoders represents its position in its coordinate reference system. The robot's speed over time R_v is obtained by computing the first derivative of the signal.

2. R_v and Tab_v are filtered to remove portions where the robot was not moving and, consequently, to separate between its positive $(+Y)$ and negative $(-Y)$ movements. To do so, the program searched for OF and speed absolute values in Tab_v and R_v respectively below the 5-th percentile of the overall values. This threshold, determined iteratively to minimize data loss while eliminating noise during stationary phases, ensures that even minimal detected movements are treated as stationary. These rows correspond to the initial and final moments of acquisition when the robot was not moving; hence, they are removed from both Tab_v and R_v (red portions in the top image of Figure 4.5). However, the resulting filtered stream may be slightly different because the portions at a speed equal to zero could be longer or shorter in R_v data stream, or the time instant where the movement direction changed could be found later or before the corresponding Tab_v data stream. As a result, the filtering step introduces a temporal shift between the two signals that must be corrected.

3. To find the instant when the robot changes its moving direction (from $+Y$ to $-Y$), the software scans Tab_v and R_v to select the rows where OF and robot speed values change from positive to negative. The turning point

(yellow portions in the top image of Figure 4.5) is identified as the point where the corresponding signal goes under the 5-th percentile of the values in Tab_v and R_v respectively (without the elements already filtered out from the previous step). Then, the negative stream is rectified, obtaining a signal always in the positive quadrant for both OF and robot speed values.

4. The robot's signal is sub-sampled to the same number of points as the corresponding OF signal of the experiment.

5. For each data point of the OF data stream Tab_v, the algorithm searches the temporally nearest neighbor of the corresponding robot's data stream R_v. The iterative procedure outputs several values as many points in the data streams, then computes their average T_{shift}, representing the temporal shift between the two signals. The temporal correction T_{shift} is then applied to R_v.

6. The two data streams include acceleration and deceleration components that must be removed to obtain only data corresponding to movements at constant speed. This step is applied on both Tab_v and R_v at the same time. Points to be removed are selected iteratively by removing parts at the beginning and at the end of the original curve and computing the linear regression with the obtained curve. The procedure removes the initial and final 1% of the whole curve first and iteratively increases the removal percentage up to 20% (with steps of 1%). For each curve obtained, a linear fit is computed coupled with the corresponding R^2. The result is a function of the distribution of R^2 values with respect to the percentage of removed data. The ideal constant velocity segment corresponds to the portion of the whole curve with the greatest R^2. It was experimentally found that the ideal value for sub-sampling is 16% for all acquisitions since R^2 does not change notably afterward. An example of this procedure is graphically shown in the bottom image of Figure 4.5, where the program iteratively selects portions of the data (depicted with colored bars to highlight the portion of data considered, in pairs) until the optimal portion is selected (in the figure, the black one).

At the end of this procedure, the data in each Tab_v is merged according to the robot's speed ($V_1 = 0.25$ m/s, $V_2 = 0.50$ m/s, $V_3 = 0.75$ m/s, $V_4 = 0.95$ m/s, $V_5 = 0.97$ m/s). The result is a total of 5 tables called $Data_v$, comprehending, for each ArUco marker, all the P_v captured points obtained during the experiment for that specific robot's speed, V_v ($v = 1...5$). Please note that the value of P_v depends on V_v because the quantity of captured points varies according to the robot's speed (more points for slower speeds).

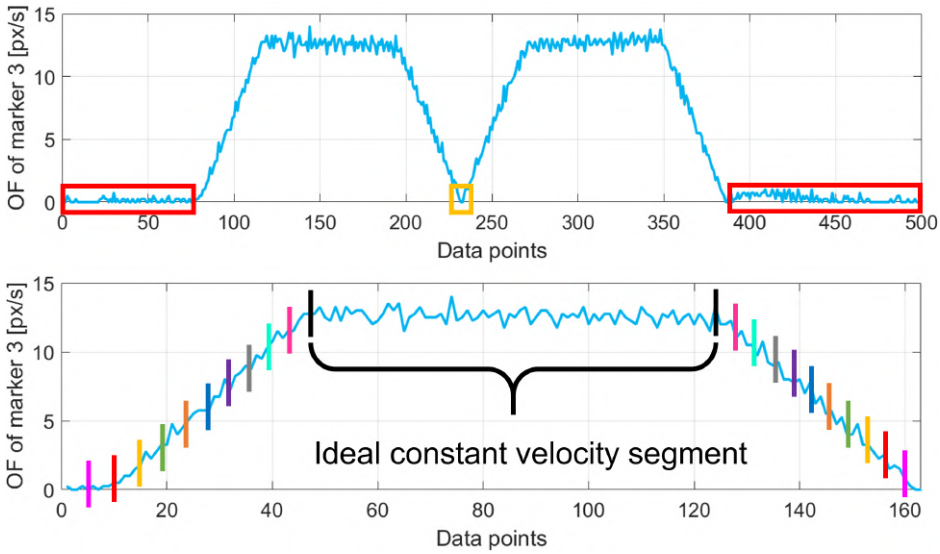


Figure 4.5: Graphical example of the filtering procedure process applied on the data of the ArUco Marker M_3 at speed $V_1 = 0.25$ m/s. (top) Removal of the portions with no robot movement (red) at the initial and final stages of the movement and corresponding to the change of direction (yellow). (bottom) Example depicting the selection process of the ideal portion of data corresponding to a movement at constant velocity.

4.4.4 Window-based filtering

The data in $Data_v$ resulting from the processing described in Section 4.4.3 is significantly noisy, especially at higher speeds. This is primarily due to vibrations of the camera and sudden environmental disturbances (e.g., light changes,

electrical noise, optical aberrations) that are inherent in real-world conditions, especially in agricultural scenarios where light scattering effects appear on the plants' canopy. These factors introduce noise and uncertainty into the acquired pictures, which in turn affect the OF output, a well-known issue in the literature [3, 4]. Moreover, in those scenarios, it is quite common to track specific objects during acquisition, for example fruits [29, 30], by using DL models such as object detectors, including in the uncertainty also the contribution of incorrect prediction or bounding box positioning. To mitigate this issue, the data in $Data_v$ is filtered again using a window-based moving average filter. This approach is also consistent with the ideal target application of agricultural fields, where the vehicle-row distance information can be less accurate (up to a few centimeters) compared to robotics or healthcare fields. In addition, this approach effectively reduces the impact of incorrect DL predictions in the case of fruit tracking.

The filter was applied iteratively on each $Data_v$ and considering the data of each ArUco marker individually. The filtered data is fitted to the analytical model in Eq. 4.2, producing an R^2 value. The optimal value of window W corresponds to the highest R^2. Best results were obtained for $W = 3$, resulting in a sub-sampling of the data corresponding to an "effective" camera acquisition speed of 20 fps (in contrast with the original 60 fps).

In the subsequent analysis, results will be shown for data with and without the application of the window-based moving average filter, for a total of $k = 1...10$ experimental models represented by tables $Data_k$. Accordingly, the quantity of captured points contained in each $Data_k$ will now be called P_k, since their number is further reduced after filtering.

4.5 Model validation and uncertainty estimation

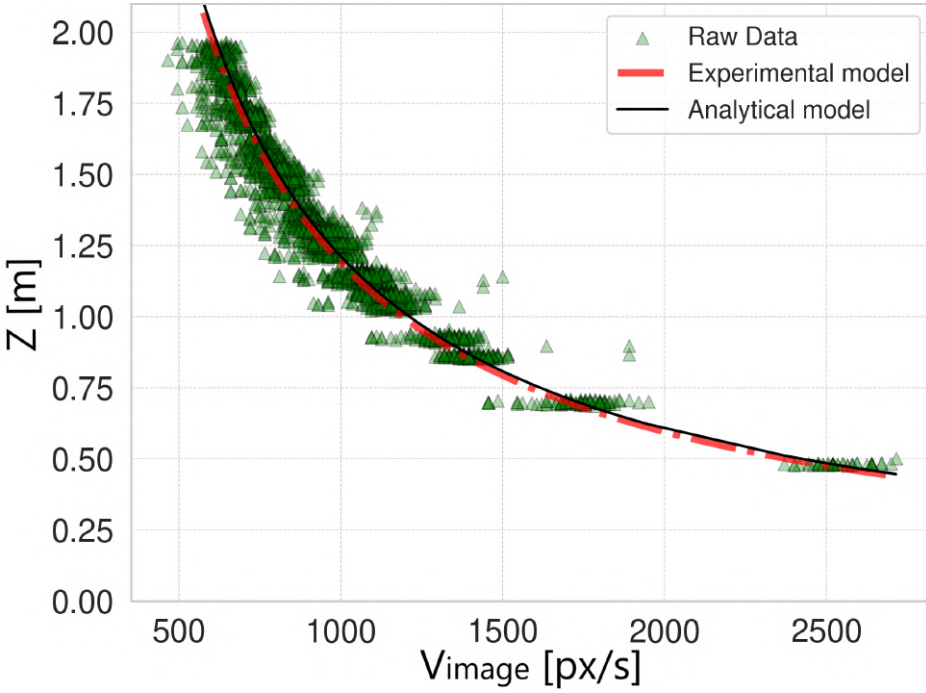
Each data set $Data_k$ contains a set of $p = 1...P_k$ points represented by a certain V_{image} obtained as the OF value of the ArUco marker m ($m = 1...5$), $OF_{i,m}$, and a certain depth value $Z_{i,m}$, computed for each acquired image i in the data set $Data_k$ as described in Section 4.4.2. For simplicity, the points of a data

set are generally represented by the pair ($V_{p,k}$, $Z_{p,k}$), a notation that takes into account all m markers points contained in a specific $Data_k$. Therefore, after conducting the pre-processing and filtering steps described in Section 4.4, the resulting $k = 1...10$ data sets $Data_k$ are used to verify if the experimental model of Eq. 4.3 is in agreement with the analytical one in Eq. 4.2. To do so, it is first necessary to obtain an estimation of the parameter K for each data set, namely K_k. For this purpose, the data set points are simply used as input for a curve-fitting method implemented in Python by the function "curve_fit" of the SciPy package [31], producing a K_k for each data set. Using the known information about the robot speed V_v (V_{camera}), the focal length of the camera (f_Y) and the acquired V_{image} of each point ($V_{p,k}$), it is straightforward to also apply the analytical model in Eq. 4.2 and compare the two.

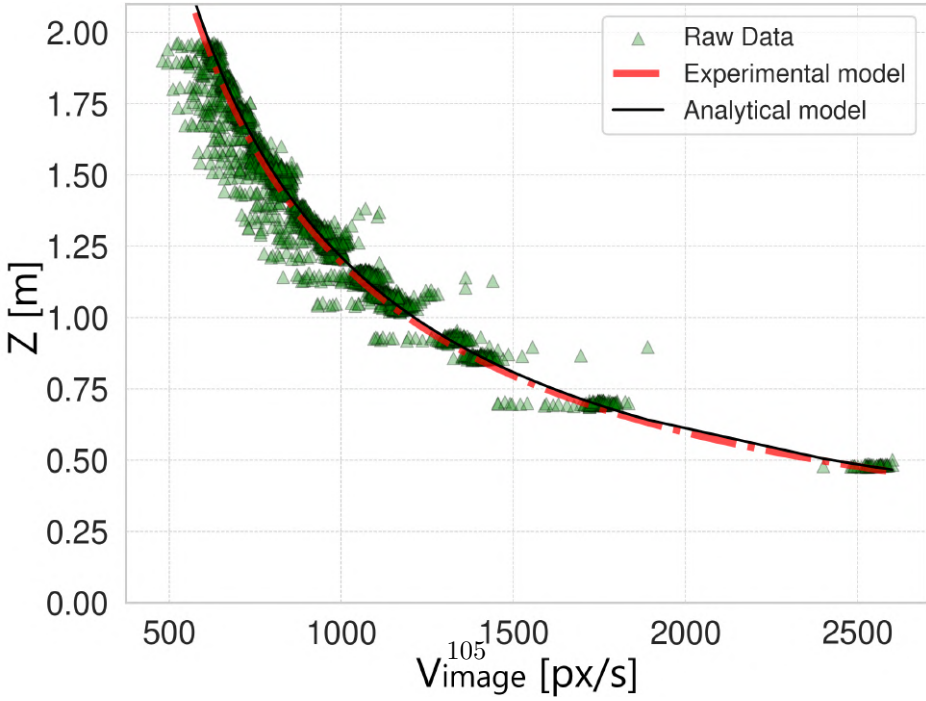
Figure 4.6 shows the resulting comparison of the two models for the original data and for the filtered data corresponding to the robot speed $V_3 = 0.75$ m/s (results for the other robot speeds are similar and are omitted for brevity). In both cases, the experimental and analytical models are overlapped (red dashed line versus black solid line), meaning that the experimental model was correctly defined and that the experimental procedure was properly conducted. However, by observing the curves, a major issue arises due to the exponential nature of the models. In fact, data points corresponding to V_{image} close to 0 px/s are distributed over a wider range of possible values, resulting in high variabilities. This effect is strongly reduced when V_{image} is higher than $500-800$ m/s according to the camera's speed, for which the model shows acceptable variability for the target application.

Due to this issue, the problem of uncertainty estimation of the experimental model is tricky to tackle. In the following analysis, two approaches are proposed to obtain the model's uncertainty: a *generalized approach* and a *complete approach*. A scheme of the analysis conducted to estimate the model's uncertainty is shown in Figure 4.7.

Both approaches are based on a common starting point based on a Monte Carlo generation of synthetic data points. The procedure, called "Monte Carlo generation" in Figure 4.7, is as follows:



(a)



(b)

Figure 4.6: Comparison between the analytical model (black line) and the experimental model (red line) for robot speed $V_3 = 0.75$ m/s. The experimental model is fitted to the points to obtain parameter K. The green triangles correspond to (a) the acquired marker points, and (b) the filtered points.

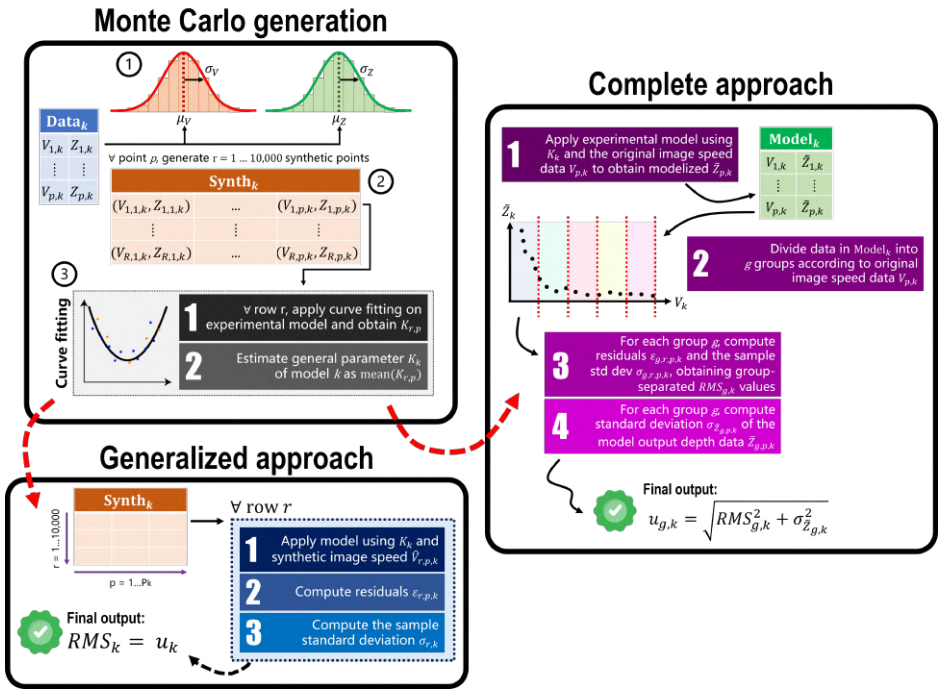


Figure 4.7: Graphical scheme of the uncertainty estimation procedure, divided into three blocks: (1) Monte Carlo generation, which is the starting point for both consequent approaches, (2) Generalized approach, giving a general uncertainty value for each tested model k, (3) Complete approach, which computes uncertainty values for each tested model k by also dividing data points by velocity group, thus accounting for their different characteristics.

1. *Original data distributions definition.* In real-world applications, the values of each point's speed and depth are affected by uncertainty. To validate the proposed experimental model, their variability was empirically set to $\sigma_V = 1$ px/s and $\sigma_Z = 0.005$ m, respectively. These two values were estimated considering the overall data acquired and the ArUco markers documentation [25, 26, 27]. Now, considering a certain set of points $Data_k$, for each point ($V_{p,k}$, $Z_{p,k}$) contained in it, two Gaussian distributions were built using the data point's actual values as the mean, μ_V and μ_Z, and the variabilities defined before, σ_V and σ_Z, as the distribution's amplitude.

2. *Synthetic data generation.* From the distributions of V and Z, a total of $10,000$ synthetic data points ($\hat{V}_{p,k}$, $\hat{Z}_{p,k}$) were generated for each original point. This process produces a table $Synth_k$ composed of $10,000$ rows and P_k columns. The synthetic data generation procedure was repeated for each tested robot's velocity V_v, for both original and filtered data, obtaining a total of 10 tables $Synth_k$.

3. *Estimation of parameter K.* Each row $r = 1...10,000$ of $Synth_k$ now contains P_k synthetic points. Therefore, it is possible to estimate parameter K for that specific row, $K_{r,k}$, by fitting the experimental model to the data in the row. Repeating this process for all rows produces $10,000$ estimated values $K_{r,k}$. Then, the overall K_k parameter that approximates the experimental model represented by the data of $Synth_k$ is obtained as the mean of all the $10,000$ $K_{r,k}$.

4.5.1 Uncertainty estimation using the generalized approach

The following steps are conducted after the "Monte Carlo generation" procedure described in Section 4.5 and are graphically shown in Figure 4.7 in the block called "Generalized approach".

Given the 10 tables $Synth_k$ and the corresponding K parameter of the model K_k, for each point $p = 1...P_k$ in the row $r = 1...10,000$, it is possible to compute $\tilde{Z}_{r,p,k}$, which is the depth value output produced by applying the experimental model in Eq. 4.3. Then, residuals $\varepsilon_{r,p,k}$ of each point are calculated as the

absolute difference between the synthetic depth $\hat{Z}_{r,p,k}$ and the depth obtained from the experimental model, $\tilde{Z}_{r,p,k}$:

$$\tilde{Z}_{r,p,k} = \frac{K_k}{\hat{V}_{r,p,k}} \tag{4.4}$$

$$\varepsilon_{r,p,k} = \left| \hat{Z}_{r,p,k} - \tilde{Z}_{r,p,k} \right| \tag{4.5}$$

where $\hat{Z}_{r,p,k}$ and $\hat{V}_{r,p,k}$ are the depth and the image speed of a synthetic data point p in row r, respectively, and K_k is the experimental model parameter estimated for each data set $Data_k$ at the end of the "Monte Carlo generation" procedure.

Then, for each row r, it is possible to compute the sample standard deviation $\sigma_{r,k}$ as:

$$\sigma_{r,k} = \sqrt{\frac{\sum_p^P \varepsilon_{r,p,k}^2}{P-1}} \tag{4.6}$$

where P corresponds to the original number of data points P_k in the columns of table $Synth_k$.

Finally, we compute the root mean square (RMS) of all the $\sigma_{r,k}$ values as:

$$RMS_k = \sqrt{\text{mean}(\sigma_{r,k}^2)} \tag{4.7}$$

This value is considered the final uncertainty for each model, $u_k = RMS_k$.

4.5.2 Uncertainty estimation using the complete approach

The following steps are conducted after the "Monte Carlo generation" procedure described at the start of Section 4.5 and are graphically shown in Figure 4.7 in the block called "Complete approach".

After generating the 10 tables $Synth_k$ and obtaining their corresponding K_k, the experimental model in Eq. 4.3 is now estimated for each velocity V_v, for

a total of 10 models. Using as input the estimated K_k and the original image velocity data of each point $V_{p,k}$ (given from the OF), the model outputs a depth value $\tilde{Z}_{p,k}$. Repeating this step for all the points in all $Data_k$ tables produces new data tables $Model_k$ in which each point is described as the pair ($V_{p,k}$, $\tilde{Z}_{p,k}$).

As already discussed at the start of Section 4.5, given the exponential nature of the model, the data points for which V was close to 0 px/s demonstrated a behavior very different from those belonging to the latter part of the graph where V was close to 3000 px/s. A workaround for this issue is the division of data points into groups according to their $V_{p,k}$ value. For all $Model_k$ tables, the data points were divided into groups according to the value of $V_{p,k}$. By considering the range of possible values of $V_{p,k}$ for that specific $Model_k$, groups were created with a step of 30 px/s (e.g. group 1 contained points with $V_{p,k}$ in range [10, 40) px/s, group 2 with $V_{p,k}$ in range [40, 70), etc.). This resulted in a certain number of groups $g = 1...G$ according to the specific table $Model_k$ considered.

The RMS_k values described by Eq. 4.7 are now calculated on the data belonging to each group g for each model k; namely, $\sigma_{g,r,k}$ obtained by using Eq. 4.6 and $\varepsilon_{g,r,p,k}$ as in Eq. 4.4. This produces group-separated RMS values for the specific experimental model k considered, $RMS_{g,k}$.

The final uncertainty in this approach is composed of two contributions for each model k and it is separated by velocity group g: one is given by the $RMS_{g,k}$, and the other is given by the standard deviation $\sigma_{\tilde{Z}_{g,k}}$ of the depth values obtained by applying the experimental model and divided by group, namely $\tilde{Z}_{g,p,k}$. The computation of the group-separated uncertainty for each model k is given by the following:

$$u_{g,k} = \sqrt{RMS_{g,k}^2 + \sigma_{\tilde{Z}_{g,k}}^2} \tag{4.8}$$

4.6 Results and discussion

The resulting uncertainties for the generalized approach are shown in Table 4.1 for both original and filtered data. The effect of the window-based filter is evident

Table 4.1: Summary of resulting generalized uncertainty for all tested robot speeds V_v. Data is shown for both the original data (u_{1-5}) and for the data resulting from the window-based filtering procedure (u_{6-10}).

Robot speed [m/s]	$\mathbf{u_{1-5}}$ **[m]**	$\mathbf{u_{6-10}}$ **[m]**
0.25	0.15	0.07
0.50	0.08	0.04
0.75	0.09	0.07
0.95	0.20	0.19
0.97	0.22	0.21

for V_1 and V_2 for which the overall uncertainty is reduced by 50%, while for V_3 the effective uncertainty reduction is only 20%. In the case of V_4 and V_5 the effect is even less evident, reducing the uncertainty of just 0.01 m in both cases (5%). This effect is related to the number of frames acquired according to the robot's speed; in fact, since the camera's acquisition rate is the same in all tests (60 fps), at lower speeds, more images are acquired representing the same scene; thus, there is not enough displacement in between two consecutive pictures for OF to work well. By sub-sampling the data, the effect is to virtually reduce the camera's acquisition rate to 30 fps, incrementing the spatial difference between two consecutive frames and thus improving OF estimation. In addition, after applying the filtering, the overall uncertainty for $V_1 = 0.25$ m/s is the same as the one obtained for $V_3 = 0.75$ m/s. Generally, best-case scenarios are obtained for $V_2 = 0.50$ m/s and for $V_3 = 0.75$ m/s.

As for the complete approach, the results of $RMS_{g,k}$ and $\sigma_{\tilde{Z}_{g,k}}$ are shown in Figure 4.8a and Figure 4.8c, respectively, for models with $k = 1...5$ (no filtering), and in Figure 4.8b and Figure 4.8d, respectively, for models with $k = 6...10$ (filtering applied). Obviously, the groups of each model are not comparable since they depend on the numerosity of points belonging to the group, which in turn depends on the total number of points P_k in the original data set $Data_k$. In Figure 4.8e and Figure 4.8f it is displayed the contribution of $RMS_{g,3}$ and $\sigma_{\tilde{Z}_{g,3}}$ towards the computation of the total uncertainty $u_{g,3}$, for $V_3 = 0.75$ in both the unfiltered and filtered cases (models with $k = 3$ and $k = 8$, respectively). The results for the other robot speeds V_1, V_2, V_4, and V_5 are similar, so they were

omitted for brevity.

It is evident that almost all the contribution is due to $RMS_{g,k}$, and its trend indicates that estimating the depth of a given point from OF is not robust in the first area of the graph where the speed of the point (OF) is lower than $500 - 800$ px/s according to the camera's speed. However, for points' speed higher than this value, the overall uncertainty is reduced to less than 20 cm. Moreover, measurement uncertainty on the computation of depth is even less than 10 cm for robot speeds equal to $V_2 = 0.50$ m/s and $V_3 = 0.75$ m/s, while it increases for the other three. This is interesting because it highlights that moving at a very low speed ($V_1 = 0.25$ m/s $= 0.9$ km/h) gives similar uncertainty values to those obtained when moving at higher speeds ($V_4 = 0.95$ m/s $= 3.4$ km/h and $V_5 = 0.97$ m/s $= 3.5$ km/h). This effect can be explained by how depth is estimated from OF. When the robot moves too slowly (V_{camera} is too low), there is not enough pixel difference between consecutive pairs of images for OF to produce an accurate estimation of $V_{image} = OF_{i,m}$, while the accuracy of the ArUco markers apparent depth $Z_{i,m}$ is higher with lower uncertainty due to a more stable image frame.

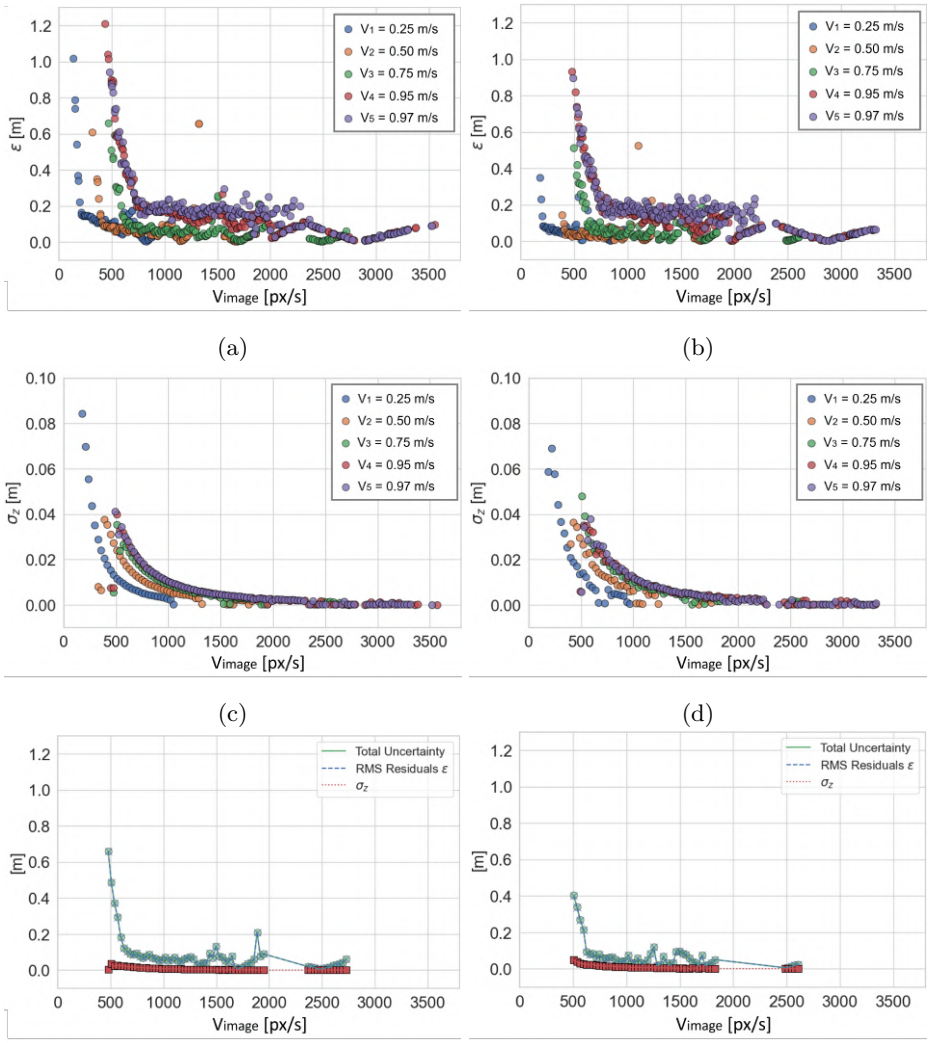


Figure 4.8: (a-b) Graphs showing the $RMS_{g,k}$ values for models with $k = 1...5$ (no filtering) and with $k = 6...10$ (filtering applied), respectively. (c-d) Graphs showing the $\sigma_{\bar{Z}_{g,k}}$ values for models with $k = 1...5$ (no filtering) and with $k = 6...10$ (filtering applied), respectively. (e-f) Graphs showing, for $V_3 = 0.75$ m/s, the contribution of $RMS_{g,3}$ (blue dashed line with 'X' markers) and $\sigma_{\bar{Z}_{g,3}}$ (red dotted line with 'squared' markers) towards the computation of the total uncertainty $u_{g,3}$ (green solid line with 'circular' markers), for $k = 3$ (no filtering) and $k = 8$ (filtering applied), respectively.

On the other hand, when the robot moves faster, the estimation of V_{image} improves until the amount of pixel difference between consecutive images is too high for OF to produce a valid estimation since the two images could be so different from each other that the point matching fails. Consequently, also the accuracy in the estimation of apparent depth becomes less accurate when V_{camera} increases. Evidently, by applying the window-based filtering, the variability of OF estimation is reduced, and the overall depth estimation is better despite losing data points. This drawback is acceptable for the target application (agriculture) and other applications for which sampling 1 data point every 3 is not an issue. Moreover, the sub-sampling of data points is closely related to the camera acquisition speed (fps): given the issue of OF not being able to produce reliable outputs when the image pairs difference is too low, adopting cameras with high fps is not the best choice. On the other hand, considering the possibility of sub-sampling the acquired frames, choosing a camera with less than 30 fps could lead to similar issues. In both cases, the outcome may be similar to the model produced for V_1.

To conclude, the following examples provide some practical conclusions for agronomists and researchers aiming at conducting on-the-go depth measurements using the proposed method: (i) how to choose the correct vehicle speed given a specific camera, and (i) how to choose the correct camera for the target application given the vehicle speed (V_{image}).

4.6.1 Practical examples

Let us consider a camera with sensor size $S_X \times S_Y = 7.2 \times 5.4$ mm, pixel resolution of $PR_X \times PR_Y = 1600 \times 1200$ px, and 60 fps acquisition speed. The camera is fixed on a rigid case mounted on the side of a vehicle, so the pixels vary along the X axis of the camera (assuming negligible vibrations, so no movement along Y). From the results obtained and discussed in the previous Section, it is assumed $V_{image}^{min} = 500$ px/s. As for the maximum, V_{image}^{max}, to keep measurement uncertainty on estimated depth values acceptable for the target application, it is suggested to adopt a maximum vehicle speed equal to $V_{vehicle}^{max} = 4 \cdot V_{vehicle}^{min}$. The camera obtains a pair of frames after a time $dT = 1/\text{fps} = 1/60$ s, so the amount

of pixels that change in two consecutive images according to the chosen speed is calculated as $C_{px} = V_{image} \cdot dT$. Using the minimum and maximum image speeds, it results that $C_{px}^{min} \approx 8$ px and $C_{px}^{max} \approx 33$ px.

Considering the target application of an agriculture scenario where a tractor moves in between two rows, it is known that inter-row distance is typically set to 2 m, so it can be reasonably assumed that the working distance WD, which is the camera-object distance, is $WD \approx 800$ mm. By using the following formula, we can calculate the camera's field of view (FoV) along the X axis:

$$FoV_{mm} = \frac{S_X \cdot WD}{f} \tag{4.9}$$

Where f is the focal length of the chosen optics. Several options exist on the market, and the choice depends on the magnification effect desired according to the operating WD, recalling from the basics of digital photography [32] that higher values of f correspond to a higher magnification and a narrow FoV, while lower values correspond to the opposite case. In this example, let us assume $f = 8$ mm, corresponding to $FoV_{mm} = 720$ mm. Now it is possible to calculate the millimeters-to-pixels resolution ratio R_s as:

$$R_s = \frac{FoV_{mm}}{PR_X} \tag{4.10}$$

A low value of R_s corresponds to a higher number of pixels needed to represent a millimeter, thus producing image details with a higher number of pixels. Having a sufficient number of pixels to represent the smallest target object in the image is key to successful computer vision applications. For this example, it results $R_s = 0.45$ mm/px. Then, the minimum and maximum vehicle speeds can be obtained using the following formulas:

$$V_{vehicle}^{min} = R_s \cdot V_{image}^{min} \tag{4.11}$$

$$V_{vehicle}^{max} = R_s \cdot 4 \cdot V_{image}^{min} \tag{4.12}$$

This results in $V_{vehicle}^{min} = 225$ mm/s $= 0.23$ m/s $= 0.81$ km/h and $V_{vehicle}^{max} = 0.90$ m/s $= 3.24$ km/h.

In the second scenario, the vehicle speed and the working distance are already defined and the question is about the choice of the right camera for the application. For the sake of the example, let us consider $V_{vehicle} = 5$ km/h $= 1.39$ m/s $= 1388.89$ mm/s and $WD = 1000$ mm. Hence, by inverting Eq. 4.11 and considering $V_{image}^{min} = 500$ px/s, the result is $R_s = 2.78$ mm/px. Let us assume that, for the target application, at least $FoV_{mm} = 1000$ mm need to be viewed by the camera. Inverting Eq. 4.10 results in $PR_X = 360$ px. The end-user is now encouraged to search among the plethora of available cameras on the market with a pixel resolution along X at least equal to 360 px, a requirement easily fulfilled nowadays. After selecting the best product for its needs, it is straightforward to also select the optics by using Eq. 4.9 to find the value of f. Once again, the resulting value is the optimal one resulting from mathematical formulations; thus, it may differ from what is available on the market. The process of finding the right camera and optics may be a long one since it requires adjustments and comparisons with several products.

Finally, two considerations should be made about possible aberrations appearing in the acquired frames when using cameras for on-the-go acquisitions. The first issue is *motion blur*, an effect appearing when the vehicle moves too fast and the camera acquires too slowly. In this case, it is recommended to use cameras with high fps. The second issue is about the camera's shutter. Low-cost cameras are typically equipped with a rolling shutter; however, this results in the upper and bottom portions of the image corresponding to two different scenes, an issue especially for fast movements (a typical example is the picture of a fan). To avoid this issue, it is recommended to choose cameras equipped with a *global shutter*.

4.7 Conclusions

The presented work deals with the topic of depth estimation from a moving monocular 2D camera leveraging optical flow. Starting from the classic analytical model used to estimate depth from moving images, an experimental model that is easier to apply for end-users is proposed and validated. The experimental set-up comprises a robot that simulates the moving vehicle, on which the camera is mounted. The target measurand is a rigid frame with 5 bars of different lengths mounted on it to simulate objects positioned at different depths. On top of each bar, an ArUco marker was fixed. A total of 5 experiments were conducted by actuating the robot at 5 speeds, each time recording a video that contains both positive and negative robot's motions. The developed software analyzes the videos to extract the ArUco markers' apparent depth (Z) and compute the optical flow (V_{image}) from pairs of images. A window-based moving average filter was applied to the acquired data to reduce noise and improve the final uncertainty.

The core of this work stands in the metrological validation of the experiments and the computation of uncertainty, for which two approaches are proposed: the generalized approach and the complete approach. In the case of the generalized approach, the best-case scenario is obtained for robot speeds equal to $V_2 = 0.50$ m/s and $V_3 = 0.75$ m/s, for which the corresponding uncertainty on depth estimation is $u_2 = 0.08$ m (no filter) and $u_7 = 0.04$ m (filter applied) for V_2, and $u_3 = 0.09$ m (no filter) and $u_8 = 0.07$ m (filter applied) for V_3. The complete approach separates the depth data according to their speed V_{image} to deal with the exponential nature of the models, producing group-separated results for each model with and without filtering. Again, the best case scenario is obtained for robot speed equal to V_2 or V_3.

Another interesting conclusion useful for end-users is that the experiments highlighted that low image speeds increase the total uncertainty when the pixel displacement between two consecutive images is insufficient, thus reducing the robustness of the optical flow algorithm. This effect directly impact both the vehicle speed and the camera's acquisition rate. Generally, it is shown that for image speeds higher than $500-800$ px/s (according to the model corresponding to

the vehicle speed), uncertainty on depth estimation drops below 0.2 m. This outcome is especially useful for applications such as on-the-go depth measurements in scenarios where 10 to 20 m uncertainty is acceptable, such as agriculture. In fact, in this specific context, it is common to have low-cost cameras mounted on moving vehicles and the main questions are: "Given a certain camera, at which speed should the vehicle move to get stable depth readings?" and "Which camera should be bought if the vehicle moves at a certain speed?". To answer these questions, two practical examples are presented and discussed, showing how the proposed work can be effectively employed by end-users in the two most common scenarios.

To conclude, the presented work is a stepping stone towards the development of reliable easy-to-use and low-cost embedded measuring systems suitable for in-field measurements for a plethora of applications, especially in agriculture. Future works will be devoted to the optimization of the presented methodology by combining it with modern Deep Learning models.

Bibliography

[1] Berthold K.P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1):185–203, 1981.

[2] Bruce D Lucas and Takeo Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *IJCAI'81: 7th international joint conference on Artificial intelligence*, volume 2, pages 674–679, Vancouver, Canada, August 1981.

[3] Stefano Savian, Mehdi Elahi, and Tammam Tillo. *Optical Flow Estimation with Deep Learning, a Survey on Recent Advances*, pages 257–287. Springer International Publishing, Cham, 2020.

[4] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12:43–77, 1994.

[5] Jun Wu, Shaoyu Liu, Zirui Wang, Xiaoyu Zhang, and Runxia Guo. Dynamic depth estimation of weakly textured objects based on light field speckle projection and adaptive step length of optical flow method. *Measurement*, 214:112834, 2023.

[6] Narjes Benameur, Tarek Kraim, Younes Arous, and Najemeddine Benabdallah. The assessment of left ventricular function in mri using the detection of myocardial borders and optical flow approaches: A review. *International Journal of Cardiovascular Practice*, 2(4):73–75, 2017.

[7] Haiyang Chao, Yu Gu, and Marcello Napolitano. A survey of optical flow techniques for robotics navigation applications. *Journal of Intelligent Robotic Systems*, 73:361–372, 2014.

[8] D.H. Diamond, P.S. Heyns, and A.J. Oberholster. Accuracy evaluation of sub-pixel structural vibration measurements through optical flow analysis of a video sequence. *Measurement*, 95:166–172, 2017.

[9] Shogo Nagano, Shogo Moriyuki, Kazumasa Wakamori, Hiroshi Mineno, and Hirokazu Fukuda. Leaf-movement-based growth prediction model using op-

tical flow analysis and machine learning in plant factory. *Frontiers in Plant Science*, 10, 2019.

[10] Piotr E. Srokosz, Marcin Bujko, Marta Bocheńska, and Rafał Ossowski. Optical flow method for measuring deformation of soil specimen subjected to torsional shearing. *Measurement*, 174:109064, 2021.

[11] Rene Ranftl, Vibhav Vineet, Qifeng Chen, and Vladlen Koltun. Dense monocular depth estimation in complex dynamic scenes. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4058–4066, Los Alamitos, CA, USA, 2016. IEEE Computer Society.

[12] Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar, Mohammad Norouzi, Deqing Sun, and David J. Fleet. The surprising effectiveness of diffusion models for optical flow and monocular depth estimation, 2023.

[13] Katrin Lasinger, René Ranftl, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer, 2019.

[14] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth, 2023.

[15] Zhenyu Li, Shariq Farooq Bhat, and Peter Wonka. Patchfusion: An end-to-end tile-based framework for high-resolution monocular metric depth estimation, 2023.

[16] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation, 2024.

[17] Oisin Mac Aodha, Ahmad Humayun, Marc Pollefeys, and Gabriel J. Brostow. Learning a confidence measure for optical flow. *IEEE transactions on pattern analysis and machine intelligence*, 35:1107–1120, 2013.

[18] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML. Evaluation of measurement data — Guide to the expression of uncertainty in measurement. Joint Committee for Guides in Metrology, JCGM 100:2008, 2008.

[19] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML. Guide to the expression of uncertainty in measurement — Part 6: Developing and using measurement models. Joint Committee for Guides in Metrology, JCGM GUM-6:2020, 2020.

[20] Vignesh Raja Ponnambalam, Marianne Bakken, Richard J. D. Moore, Jon Glenn Omholt Gjevestad, and Pål Johan From. Autonomous crop row guidance using adaptive multi-roi in strawberry fields. *Sensors*, 20(18), 2020.

[21] Onur Ozyesil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion, 2017.

[22] Bernardo Lanza. Depth estimation from optical flow in agricultural fields. https://github.com/bernardolanza93/DepthFromOpticalFlow.git, 2024.

[23] Alessandro Umbrico, Andrea Orlandini, Amedeo Cesta, Marco Faroni, Manuel Beschi, Nicola Pedrocchi, Andrea Scala, Piervincenzo Tavormina, Spyros Koukas, Andreas Zalonis, Nikos Fourtakas, Panagiotis Stylianos Kotsaris, Dionisis Andronas, and Sotiris Makris. Design of advanced human–robot collaborative cells for personalized human–robot collaborations. *Applied Sciences*, 12(14), 2022.

[24] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[25] S. Garrido-Jurado, R. Muñoz-Salinas, F.J. Madrid-Cuevas, and M.J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

[26] S. Garrido-Jurado, R. Muñoz-Salinas, F.J. Madrid-Cuevas, and R. Medina-Carnicer. Generation of fiducial marker dictionaries using mixed integer linear programming. *Pattern Recognition*, 51:481–491, 2016.

[27] Francisco J. Romero-Ramirez, Rafael Muñoz-Salinas, and Rafael Medina-Carnicer. Speeded up detection of squared fiducial markers. *Image and Vision Computing*, 76:38–47, 2018.

[28] OpenCV Official Documentation Ana Huamán. Changing the contrast and brightness of an image, 2016. Online. Accessed: May 2024. URL: https://docs.opencv.org/4.x/d3/dc1/tutorial_basic_linear_transform.html.

[29] Jordi Gené-Mola, Marc Felip-Pomés, Francesc Net-Barnés, Josep-Ramon Morros, Juan C. Miranda, Jaume Arnó, Luís Asín, Jaume Lordan, Javier Ruiz-Hidalgo, and Eduard Gregorio. Video-based fruit detection and tracking for apple counting and mapping. In *2023 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pages 301–306, 2023.

[30] Mar Ferrer-Ferrer, Javier Ruiz-Hidalgo, Eduard Gregorio, Verónica Vilaplana, Josep-Ramon Morros, and Jordi Gené-Mola. Simultaneous fruit detection and size estimation using multitask deep neural networks. *Biosystems Engineering*, 233:63–75, 2023.

[31] Scipy official documentation. Curve fit function of scipy optimize library, 2022.

[32] D A Rowlands. *Physics of Digital Photography (Second Edition)*. 2053-2563. IOP Publishing, 2020.

Chapter 5

3D Reconstruction of Plants and Digital

5.1 3D Reconstruction of Plants and Digital Twin

Localization and depth estimation pave the way for detailed 3D reconstruction, an essential component in modern agricultural monitoring. By combining spatial data with depth information, it becomes possible to generate comprehensive 3D models of plants. These models not only support advanced yield analysis and resource allocation but also form the basis for developing digital twins—virtual replicas that enable continuous monitoring and decision-making. This section investigates the methodologies and applications of 3D reconstruction, with a focus on creating reliable, field-validated systems for agricultural environments.

5.1.1 DIGIFRUIT and Research Activities in Lleida

The DIGIFRUIT project, funded by the Spanish Ministry of Science and Innovation, aims to accelerate the adoption of Precision Agriculture by developing low-cost orchard monitoring systems. The project integrates photonic sensors such

as RGB-D cameras and LiDAR into mobile platforms, enabling 3D reconstruction of orchards to support applications like canopy geometry measurement, yield estimation, and site-specific management practices [1]. The focus on affordable technology ensures that these solutions are accessible to a wide range of agricultural stakeholders, promoting digital transformation in the sector. During my research stay at the Universitat de Lleida, I collaborated closely with the GRAP research group and contributed to various aspects of the DIGIFRUIT project. This period allowed for practical validation of methodologies in controlled and real-world agricultural environments. Specifically, my work included:

- **Experimental Campaigns:** Conducted field tests in collaboration with local companies and the university's experimental orchards. Initial tests focused on evaluating the influence of lighting conditions on photonic sensors, while subsequent tests addressed the integration of sensors into mobile platforms for dynamic 3D orchard reconstruction.
- **Collaboration with Industry:** Coordinated efforts between the university and agricultural enterprises to test sensor systems in commercial orchards located in Gimenells (LLeida, ES). These trials validated the system's applicability in diverse environmental conditions and provided valuable insights for real-world implementation.
- **Paper Contributions:** Contributed to publications focusing on metrological validation and algorithmic development for 3D reconstruction. These works highlighted the integration of low-cost sensors and advanced data processing methodologies in agricultural contexts.

The experimental campaigns were divided into two main stages:

1. **Illumination Testing:** Conducted in the university's experimental orchard, this phase assessed how varying light conditions affect RGB-D and LiDAR sensor performance. These tests provided essential data for optimizing sensor configurations.
2. **3D Data Acquisition:** Performed in collaboration with local companies in Gimenells, these tests focused on generating point clouds for orchard

reconstruction. Using RGB-D cameras and GNSS-equipped platforms, the trials evaluated the system's ability to create accurate and comprehensive 3D models in real-world conditions.

These activities laid the foundation for subsequent research on 3D orchard reconstruction and the development of an embedded system for low-cost agricultural monitoring. By addressing the technical challenges of sensor integration and validating the methodologies in diverse environments, the project advances the field of Precision Agriculture and establishes a scalable framework for future research.

5.2 Illumination Testing

Lighting conditions significantly influence the performance of photonic sensors, impacting the quality and reliability of data acquisition in agricultural environments. Experiments conducted in Lleida evaluated the effect of ambient illumination on the performance of RGB-D cameras and LiDAR sensors under realistic field conditions:

- **RGB-D Cameras**: Demonstrated a strong dependence on lighting conditions, with accuracy improving as ambient light decreases. Under low light levels (< 1000 lux), RGB-D cameras achieved point densities comparable to those of LiDAR sensors.
- **LiDAR Sensors**: Showed robust performance across varying lighting conditions, maintaining consistent accuracy regardless of ambient light.

These findings highlight the critical role of controlled illumination for achieving high-quality 3D reconstructions in field conditions. This research, which I contributed to by designing experimental protocols and conducting data acquisition, has been detailed in the paper titled *"Impact of Lighting Conditions on the Performance of Photonic Sensors for 3D Orchard Reconstruction."* The paper was submitted to, accepted and reviewed by the **14th European Conference on Precision Agriculture (ECPA 2025)**, which will be held in Barcelona.

The methodologies and results presented provide a foundational understanding of how lighting influences sensor performance, guiding the development of robust and reliable monitoring systems for precision agriculture.

5.2.1 Introduction

Monitoring the geometric and structural characteristics of fruit orchards is necessary for the deployment of precision fructiculture strategies. Several non-destructive technologies, such as photogrammetry and light detection and ranging (LiDAR), have become popular for the automatic characterization of tree crops [2]. Among photogrammetric techniques, structure-from-motion (SfM) stands out for its ability to generate accurate 3D reconstructions by moving an RGB camera to different positions and taking photos with significant overlap. SfM technique has been applied for plant phenotyping in outdoor conditions [3, 4] and for in-field fruit detection [5, 6]. However, the application of these photogrammetric techniques for orchard monitoring is limited by their high computational cost and reliance on lighting conditions. LiDAR is a robust, high-performance technology well-suited for outdoor use, relying on the emission of laser light and the detection of the backscattered signal. LiDAR sensors have been extensively applied to characterize tree crops such as pears, apples, vineyards, and olives [7, 8, 9]. Despite their advantages, conventional LiDAR sensors, equipped with moving mechanical components such as mirrors to direct the laser beam, are typically expensive. This high cost limit their adoption in the sector. Nowadays, there is a wide range of RGB-D cameras available on the market that provide colour, depth, and IR intensity data at a much lower cost than conventional LiDAR sensors. However, their application in monitoring fruit orchards has been limited to date due to their poor performance under daylight conditions [10, 11, 12, 13]. Additionally, the recent advent of affordable LiDAR-based sensors, typically based on a solid-state architecture with no moving parts, opens up new opportunities. The present work assesses the performance of various RGB-D cameras and emerging low-cost LiDAR sensors in producing accurate three-dimensional reconstructions of fruit trees across varying lighting conditions. This evaluation serves as an initial step toward the development of an affordable orchard monitoring

system.

5.2.2 Materials and Methods

Field tests were performed at an experimental pear orchard located in the School of Agrifood and Forestry Engineering and Veterinary Medicine at the Universitat de Lleida (Catalonia, Spain). The canopy was trained as a fruiting wall, with a spacing of 4.5 m (char check)1.5 m, with the rows oriented in an approximately north-south direction. The study was focused on measuring a tree located in the middle of the row, 2.9 m high and 1.3 m wide.

Three RGB-D camera types were used in this experiment: time-of-flight (ToF) (Azure Kinect DK, Microsoft Corporation, Redmond, WA, USA), active stereoscopy (RealSense D455f, Intel, Santa Clara, CA, USA) and passive stereoscopy (ZED X Mini, Stereolabs, San Francisco, CA, USA). The two evaluated LiDAR sensors (Mid-360 and Mid-70) were manufactured by Livox Technology (Shenzhen, Guangdong, China) and are characterized by emitting a non-repetitive scanning pattern. The Mid-360 is based on a mirror hybrid-solid technology and features a field of view (FoV) of $360^{\underline{o}}$ x $59^{\underline{o}}$, while the Mid-70 uses rotation-free photoelectronic components and has a circular FoV of $70.4^{\underline{o}}$. Table 5.1 hows the specifications of the RGB-D cameras and LiDAR sensors under the configuration options used in the experiment.

Sensor	Type	Range (m)	Resolution	Field of View
Azure Kinect DK	ToF	0.5–3.86	1280x720 px	$90° \times 59°$
RealSense D455f	Stereo	0.6–6	640x480 px	$75° \times 65°$
ZED X Mini	Stereo	0.1–8	1920x1080 px	$87° \times 62°$
Mid-360	LiDAR	0.1–70	-	$360° \times 59°$
Mid-70	LiDAR	0.1–260	-	70.4° circular

Table 5.1: Specifications of the evaluated sensors.

As shown in Figure 5.1 sensor acquisition and data storage were managed by

a Jetson AGX Orin embedded module (NVIDIA, Santa Clara, CA, USA). The acquisition system was powered by a high energy density 12 V lithium battery with a built-in battery management system (BMS). The sensors were located at a distance of 3.8 m from the axis of the tree row and remained static (eastwards-oriented) throughout the experiment. RGB-D and LiDAR acquisitions were made every 15 minutes between 11:00 h and 19:00 h (April 11th, 2024, UTC+2) and every 5 minutes between 19:00 h and 21:00 h (rapid ambient light decrease during dusk). Three different integration times (IT) of 1 s, 5 s and 10 s were used for the LiDAR acquisitions. The ambient illuminance was measured during the entire experiment by means of an automatic lux meter (PCE-LMD 10, PCE Instruments, Meschede-Freienohl, Germany) at a sampling rate of 1 Hz. In addition, 1853 photographs of the tree were taken with a handheld EOS 60D DSLR camera (Canon Inc. Tokyo, Japan) equipped with a 5184 $\times$ 3456 pixels CMOS APS-C sensor. Regarding the camera lens, a Canon EF 50 mm f/1.4 USM was used. These images were used to create an accurate photogrammetric 3D reconstruction by applying SfM and multi-view stereo.

The photogrammetric reconstruction was used as reference to register (using rigid transformations) and segment the 3D point clouds captured by the different sensors. Once registered, a region of interest (ROI) was defined for all the point clouds. The RGB-D cameras and low-cost LiDAR sensors were evaluated in terms of resolution, precision, and accuracy following the methodology proposed by Gené-Mola et al. (2020b). Regarding resolution, the number of points and the average point cloud density were computed for each acquisition. Accuracy was assessed by calculating the average Euclidean distance between each point in the evaluated cloud and its nearest point in the reference cloud (photogrammetric reconstruction). To measure this distance, the software CloudCompare v2.13.2 (EDF R&D, Paris, France) was used. The precision or repeatability was evaluated by computing the distances between point clouds corresponding to different replicates of the same acquisition.

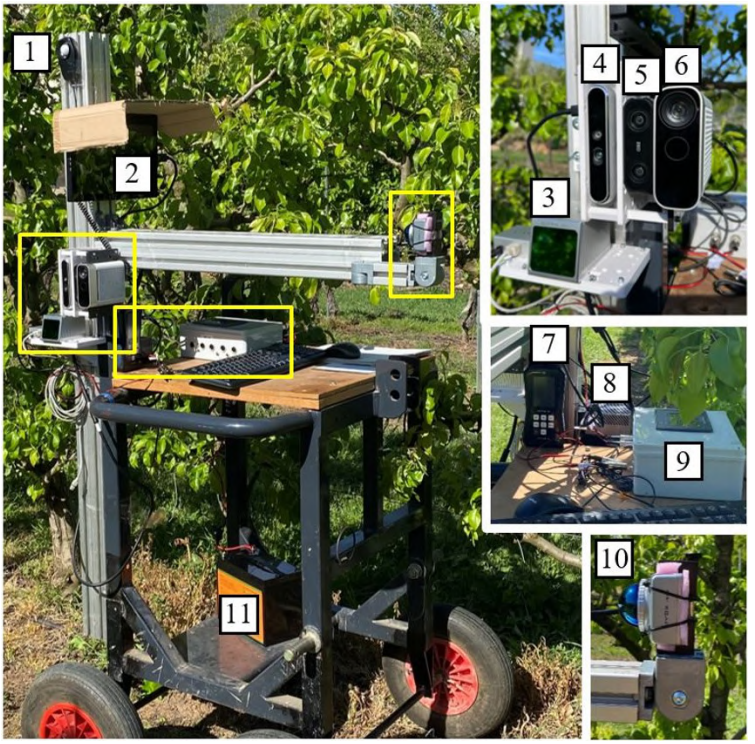


Figure 5.1: Set-up of the acquisition system. (1) Light sensor; (2) Touchscreen; (3) Livox Mid-70; (4) Intel RealSense 455f; (5) Stereolabs ZED X Mini; (6) Microsoft Azure Kinect DK; (7) PCE-LMD 10 Lux meter; (8) Jetson AGX Orin; (9) Connection box and battery monitor; (10) Livox Mid-360; (11) Lithium battery.

5.2.3 Results and Discussion

A qualitative analysis revealed that the most faithful 3D point clouds were provided by the Azure Kinect camera and by the LiDAR sensors. Therefore, only the Azure Kinect and the LiDAR sensors were analysed in the subsequent evaluations. Even so, it is noteworthy that the stereovision RGB-D cameras demonstrated robust performance under different lighting conditions, and their low power requirements make them ideal for numerous agricultural robotics applications. Figure 5.2 presents several point clouds acquired by the Azure Kinect camera and by the Mid-360 and Mid-70 LiDAR sensors. It is observed that the Azure Kinect was strongly affected by daylight. Hence, only Azure Kinect acquisitions between 19:00 h and 21:00 h (illuminance ranging from 4350 lux to 0 lux) were considered. Regarding the LiDAR sensors, it can be seen how the density of the point clouds increased with integration time.

As can be seen in Figure 5.2, the number of points of the LiDAR acquisitions increased with the integration time, but it did not depend on the lighting level. Although a similar behaviour was observed for both LiDAR sensors, the Mid-70 provided clouds with a number of points one order of magnitude higher. This difference is attributed to the fact that although both sensors have the same point rate, in the Mid-70 these points are concentrated in a smaller FoV. By contrast, the Azure Kinect showed a strong direct correlation (r = 0.98) between point number and illuminance. For low lighting conditions (¡ 1000 lux), the Azure Kinect provided a number of points similar to that of the Mid360 with an IT of 5 s.

Figure 5.4 shows accuracies ranging from 10 mm to 12 mm for both LiDAR sensors, with no observed dependence on illuminance or integration time. To avoid redundancy, only the accuracies for an integration time of 5 s are presented, as they are similar to those obtained for 1 s and 10 s. Therefore, the accuracy does not seem to be improved on the Mid-70 compared to the Mid-360, despite a higher number of points. Conversely, the accuracies for the Azure Kinect were progressively improved as the ambient light decreased, achieving 7 mm for less than 1500 lux.

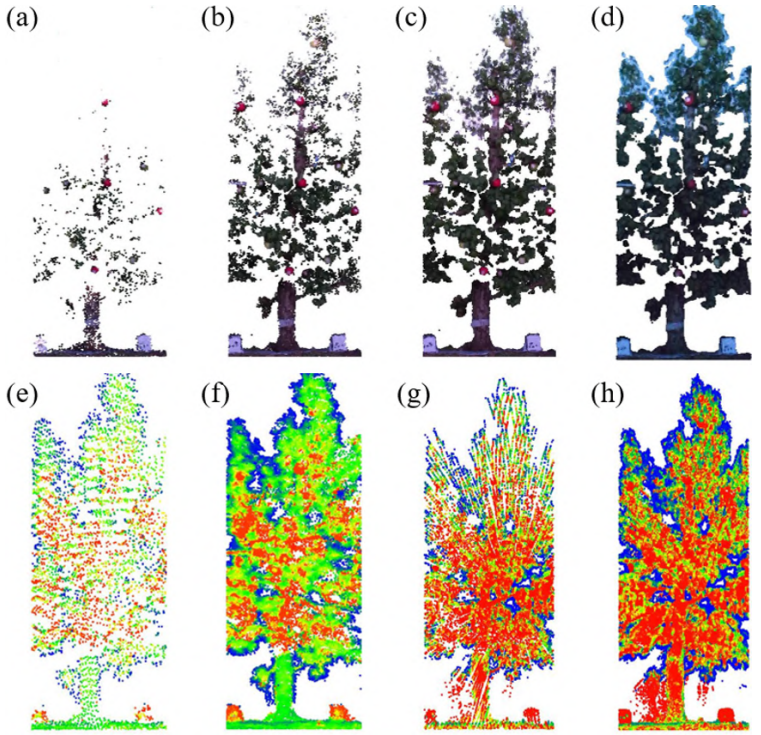


Figure 5.2: RGB-D point clouds acquired by the Azure Kinect camera under different lighting conditions: (a) 3151 lx, (b) 1798 lx, (c) 327 lx, (d) 2.5 lx. IR clouds acquired by the LiDAR sensors using different integration times (IT): (e) Mid-360 with IT = 1 s, (f) Mid-360 with IT = 10 s, (g) Mid-70 with IT = 1 s, (h) Mid-70 with IT = 10 s.

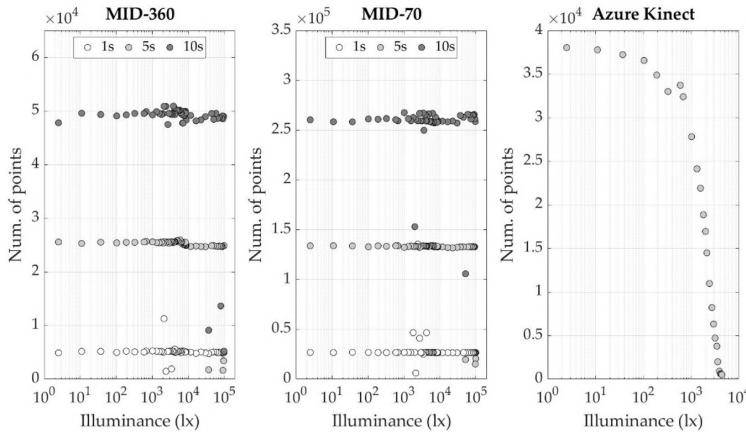


Figure 5.3: Evolution of the number of points acquired by the evaluated sensors as a function of illuminance.

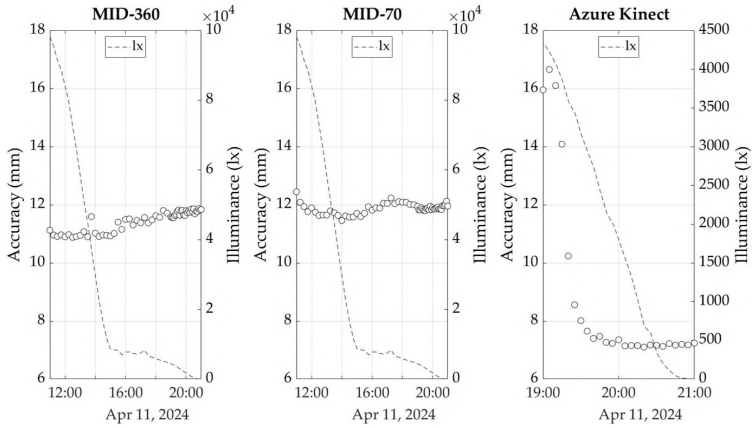


Figure 5.4: Evolution of the point cloud accuracy for the evaluated sensors throughout the experiment.

Figure 5.5 demonstrates that the precision or repeatability of both LiDAR sensors was not affected by illuminance and improved with longer integration times. Precision values of8.5 mm and 4 mm (10 s of integration) were achieved by the Mid-360 and the Mid-70, respectively. Like the other metrics, the precision of the Azure Kinect exhibited a strong dependence on the illuminance, ranging from 34 mm at 4350 lux to less than 6 mm in the absence of ambient light.

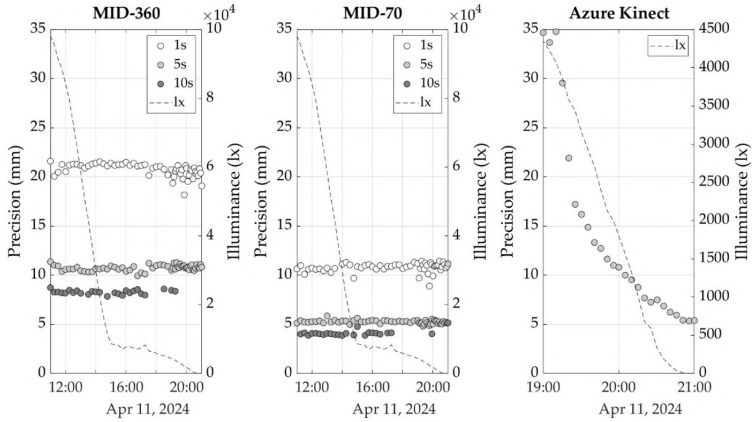


Figure 5.5: Evolution of the point cloud precision for the evaluated sensors throughout the experiment.

The evaluation of the LiDAR sensors demonstrated that both the Mid-360 and the Mid70 can generate 3D reconstructions of fruit trees with a high level of accuracy, which remained unaffected by either the integration time or the illuminance. The selection of a specific LiDAR sensor will depend on the intended application. With the sensor operating as a static terrestrial laser scanner (STLS), the Mid-70 offers advantages in terms of higher point cloud density and greater precision. On the other hand, the Mid-360 is advantageous when the objective is to scan a larger area due to its wider field of view. In addition, according to the manufacturer, the non-repetitive scanning technology used by both LiDAR sensors allows for a longer integration time to increase the FoV coverage (the total area illuminated by laser beams divided by the total area in FoV) (Livox, 2024). Regarding RGB-D cameras, the one based on the ToF principle (Azure Kinect) produced reliable reconstructions of the tree. However, this camera is

affected by sunlight and can only achieve accuracy, precision and point cloud density similar to that of LiDAR sensors when operating under reduced lighting conditions (¡ 1000 lux). Although sunlight is a major limitation for field phenotyping applications, it is important to note that in this experiment the Azure Kinect camera was positioned at a distance from the tree near to its maximum range. The authors have observed that its performance under sunlight improves significantly when the distance is reduced. However, shorter distances may result in the sensor not capturing the entire tree, depending on its height. The Azure Kinect provides both colour and depth data, which can be advantageous in applications where, in addition to the geometric and structural characterization of trees, segmentation and/or detection of vegetative organs (fruits, branches, etc.) is required.

5.2.4 Conclusions

The tested low-cost LiDAR sensors presented accuracy and precision values that makes them suitable for tree crops characterization. Most importantly, its behaviour is robust to changes in lighting conditions. For its part, the Azure Kinect (ToF RGB-D camera) is an alternative that provides additional colour data with similar point cloud densities, accuracy, and precision. However, its performance was significantly affected by sunlight. As future work, these low-cost sensors will be mounted on a mobile terrestrial platform to complete 3D reconstructions of fruit orchards. The availability of cost-effective systems for monitoring fruit orchards could accelerate the adoption of these technologies by the sector, while also contributing to the reduction of fertilizer, pesticide, and water usage through a deeper understanding of the plantations and their variability.

5.2.5 Acknowledgements

This work was partially funded by the Spanish Ministry of Science and Innovation, AEI/10.13039/501100011033, European Union NextGeneration, PRTR (grant number TED2021-131871B-I00 [DIGIFRUIT project]) and by the European Union, FSEREACT-EU, PON “Research and Innovation 2014–2020”, D.M.

1061/2021, contract number DOT1346224-8

5.3 SLAM for 3D Orchard Reconstruction

Following the illumination experiments, the next phase of the research focused on integrating the validated sensor configurations into a mobile system for in-field 3D orchard reconstruction. Leveraging the outcomes of the illumination tests, we not only refined our understanding of sensor performance under various lighting conditions but also obtained crucial insights regarding optimal sensor placement—for instance, the ideal distance from tree canopies and other practical configuration parameters. These findings informed the design of our mobile setup, ensuring that sensor positions and orientations were strategically chosen to maximize data quality in a dynamic agricultural environment. The integrated approach paved the way for a robust, mobile hardware configuration, capable of delivering high-precision 3D reconstructions of orchard environments.

The following sections will detail the setup of the sensors and the core hardware components that form the backbone of the system, providing an in-depth look at both the sensor integration strategy and the overall architecture of the mobile platform.

This research was presented at the MetroAgriFor 2024 conference under the title *Metrological Assessment of RGB-D Cameras for 3D Orchard Reconstruction*, highlighting its contribution to the field of precision agriculture and metrology.

Core Hardware Components

The system integrated several advanced components, ensuring robust data acquisition and processing:

- **NVIDIA Jetson AGX Orin:** A high-performance embedded module providing GPU-accelerated computing for real-time processing of RGB-D data and 3D point clouds.

- **Microsoft Azure Kinect DK:** A time-of-flight (ToF) RGB-D camera capable of producing high-resolution depth images (640 × 576 px) and RGB images (1920 × 1080 px), optimized for narrow field-of-view (NFOV) applications. The camera was used to extract point clouds for canopy reconstruction and tree feature analysis.
- **Intel RealSense D455f:** An active stereoscopy RGB-D camera, complementing the Azure Kinect for depth and RGB imaging in challenging light conditions.
- **Livox Mid-70 and Mid-360 LiDAR Sensors:** Low-cost LiDAR units providing accurate 3D reconstructions with high point density, suitable for large-scale orchard mapping.
- **MTi-630 IMU (Xsens):** A 9-axis inertial measurement unit for tracking movement and orientation, ensuring precise alignment of the data.
- **RTK GNSS (Ardusimple RTK2B):** Provided accurate georeferencing for spatially locating the acquired data.

Power and Portability

The system was powered by a high-energy density 12V lithium battery equipped with a battery management system (BMS). This configuration ensured sustained operation in the field, with the following features:

- **Compact Integration:** The hardware was mounted on a lightweight aluminum frame fixed to an electric off-road scooter for mobile acquisition.
- **Ease of Transport:** The system was designed for portability, balancing computational power with mobility for seamless deployment in orchard environments.

Sensor Configuration and Deployment

The sensors were strategically configured for optimal orchard data acquisition:

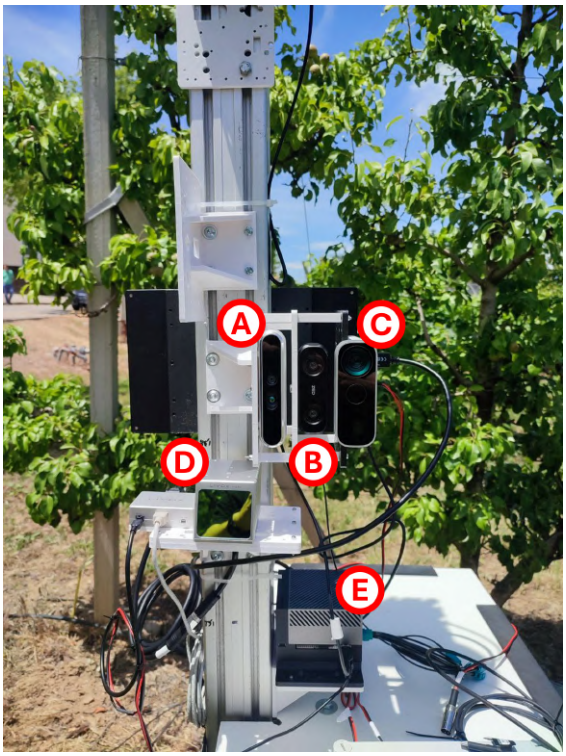


Figure 5.6: Sensors setup developed during the PhD period in Lleida. The system integrates state-of-the-art sensors, including the NVIDIA Jetson AGX Orin for GPU-accelerated computing (E), Microsoft Azure Kinect DK (C), Intel RealSense D455f (A), and ZED X (B) for RGB-D imaging and Livox LiDAR mid-70 for 3D mapping (D). Mounted on an electric off-road scooter, the setup ensures portability and high-performance data acquisition in orchard environments.

- Mounted at a fixed distance of 3.8 m from the tree axis for consistent measurements.
- Operated at speeds of up to 4 km/h, allowing for real-time data capture during traversal of orchard rows.
- Data acquisition managed via the Jetson AGX Orin, enabling seamless synchronization of multiple sensors and ensuring high-quality point cloud generation.

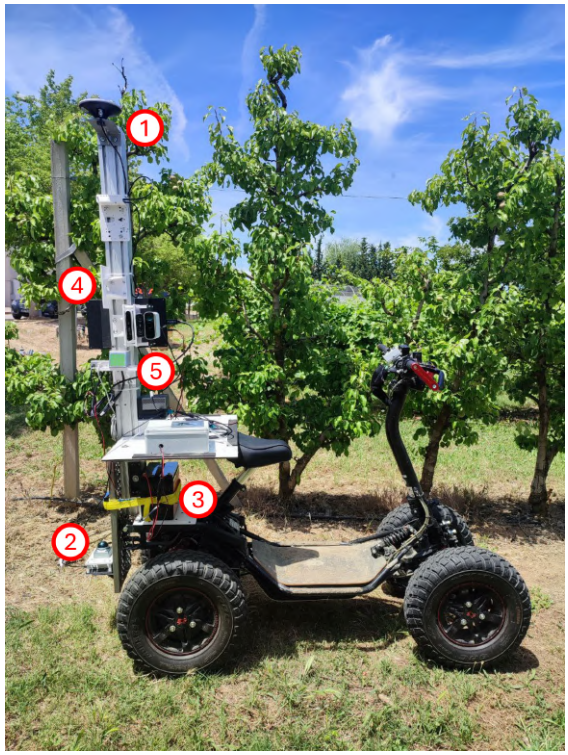


Figure 5.7: Agriculture Embedded platform overall for precise orchard mapping. The system integrates advanced sensors, including the Livox Mid-360 LiDAR (2) mounted at the base for 3D mapping, an RTK GNSS antenna (1) placed at the highest point for accurate georeferencing, and a high-energy lithium battery (3) ensuring portability and sustained field operation. Additional components include the NVIDIA Jetson AGX Orin (5) for real-time computation and synchronized acquisition, providing a versatile solution for precision agriculture tasks.

Applications and Achievements

This system was used to perform detailed orchard measurements and 3D reconstructions:

- Captured high-density 3D point clouds for analyzing canopy geometry, including tree height, volume, and trunk diameters.

- Validated the integration of RGB-D and LiDAR sensors for simultaneous depth and RGB imaging, enabling multi-sensor fusion.
- Enabled SLAM-based trajectory tracking to improve spatial alignment and reduce errors in reconstructed data.

The system played a key role in advancing research on orchard monitoring, as documented in collaborative publications with researchers from Lleida [14] (section 5.2). It demonstrated the feasibility of combining low-cost sensors with high-performance computational platforms for agricultural applications, paving the way for scalable orchard management solutions.

Metrological Insights and Uncertainty Propagation

A central aspect of this study was the metrological validation of the reconstructed 3D models, specifically the estimation of sphere radii used as reference objects. The methodology addressed unique challenges, such as a high density of internal points within the spheres, which required careful handling to avoid skewed results. The procedure included:

1. **Center of Mass Calculation**: The centroid of the point cloud was calculated as the mean of all points.
2. **Radius Estimation**: The radial distances of all points from the centroid were computed, and the 95th percentile was used to define the estimated sphere radius, capturing the majority of the surface points while excluding outliers.
3. **Convex Hull and Volume Discrepancy**: A convex hull was constructed to approximate the actual volume of the sphere. The difference between this volume and the theoretical volume of a perfect sphere (based on the estimated radius) was treated as the volume uncertainty.
4. **Uncertainty Propagation**: Using the relationship between sphere volume and radius, the volume uncertainty was propagated to estimate the

radius uncertainty. This provided a robust confidence interval for the radius measurement.

Additional metrics, such as point density and sphericity (computed via convex hull analysis), were employed to validate the accuracy and repeatability of the system. The fusion of front and back views of each sphere further enhanced the robustness of the measurements, ensuring comprehensive coverage and minimizing errors from occlusion.

Algorithmic Innovations and Hierarchical SLAM

The custom SLAM algorithm developed for this study was specifically tailored to handle the limitations of low-resolution RGB-D data and dynamic environmental conditions. Key features of the algorithm include:

- **Hierarchical Merging of Point Clouds**: Linear motion segments, which tend to produce higher-quality data, were prioritized in the merging process. Noisy data from turning maneuvers were processed separately to prevent error propagation.

- **Iterative Coarse-to-Fine Alignment**: This method ensured precise alignment of sequential point clouds, starting with an initial coarse alignment followed by iterative refinements to achieve high accuracy.

- **Error Isolation and Robustness**: By isolating data from challenging conditions (e.g., sharp turns or high vibration zones), the algorithm reduced spatial drift and improved the overall coherence of the reconstructed model.

The hierarchical approach not only addressed the challenges of sparse and noisy data but also ensured that high-quality data segments were given precedence, enhancing the reliability of the final 3D models.

5.3.1 Materials

Experimental Set-up

Data acquisition was performed on 31 May 2024, at an experimental apple orchard (Tutti® Hot84A1') located in Gimenells, Catalonia, Spain (lat. 41º39'11"N, long. 0 º23'28"E, elev. 259 m) and owned by the Institut de Recerca i Tecnologia Agroalimentàries (IRTA). The trees were trained as bi-axis with a planting spacing of 3×1.2 m, maximum canopy height of 3.5 m and at a phenological growth stage BBCH 73.

Although measurements were taken throughout the plantation, this study focuses on two consecutive trees on which six target foam spheres of known radius (97 mm, 72 mm, 48.5 mm, 38.5 mm, 28 mm, and 24 mm) were hung (Fig. 5.8). The measurement area is delimited by two ranging rods with colored stripes and two metallic folding rulers horizontally oriented. The six targets serve as a reference to evaluate the accuracy of the resulting 3D point clouds.

Please note that Fig. 5.8 also shows two larger spheres used as calibration objects for the system and another sphere with a diameter of 20 mm, which was unused since it was too small.

Sensor Configuration

The RGB-D camera used in this experiment was the Azure Kinect DK (Microsoft Corporation, Redmond, WA, USA). This device integrates a time-of-flight (ToF) camera, an RGB sensor, and an inertial measurement unit. The Azure Kinect was configured in the narrow field-of-view (NFOV) unbinned operating mode, providing resolutions of 1920 × 1080 pixels and 640 × 576 pixels for the RGB and depth images, respectively [15, 16]. The sampling rate was set to acquire data at 30 Hz (maximum capacity). The camera was attached onto an aluminum mast profile situated at the rear of a four-wheel electric off-road scooter used for the field acquisitions (Fig. 5.7).

The scooter was also equipped with an RTK2B GNSS RTK system (Ardusim-

Figure 5.8: Overview of the experimental set-up mounted in the field, including (1) the calibration foam spheres, (2) the metallic folding rulers used for sensor accuracy assessment, and (3) the ranging rods with colored stripes, necessary to identify the portion of the orchard that will be used as reference.

ple Co., Ltd., Lleida, Spain) capable of delivering georeferenced data at 10 Hz and a 9-axis MTi 630 IMU (Xsens Technologies B.V., Enschede, The Netherlands). Data acquisition was performed with the vehicle moving at 4 km/h.

The second device used to obtain data from the orchard was the high-performance bMS3D-4CAM backpack mobile terrestrial laser scanner (Viametris, Louverné, France) boarded on the electric off-road scooter. This device integrates two VLP-16 LiDAR sensors (Velodyne, San Jose, CA, USA), each of which emits 16 laser beams with an acquisition rate of 300,000 points/s.

5.3.2 Methods

Data Extraction and Preprocessing

From the raw acquisition, we produce an MKV file, which can embed various types of data. We extract RGB and Depth streams from this file. Using intrinsic data, these streams are fused to generate a point cloud with 368,648 points. This operation is performed using a function from the Kinect Azure library, optimized for this data format. Colors are associated with the point cloud to help identify objects of interest.

During the point cloud extraction, we remove all-zero points to reduce its size. This process eliminates 99.5% of all points (average rejected points: 366,725 over the total 368,648 points). The high rejection rate may be due to three reasons: (i) the distance from the objects, which is 3.5 m, (ii) the lighting conditions altering the acquisition (it is well known that ToF devices are affected by light interference [16]), (iii) the fast acquisition speed.

As a result, we obtain a point cloud of about 2,000 points, which lacks detailed features, fully showing at least two plants including the ground. In addition, we further refined and filtered the point cloud using an outlier removal filter, deleting an additional 10% of the points.

However, standard SLAM algorithms cannot perform robust ICP with a low numerosity of points like in our case, especially in difficult environments. The

ICP algorithm registers the point cloud with a significant registration error and spatial drift propagation due to the lack of points. Moreover, the presence of the ground in the point cloud is a noise source because, since it contains most of the point cloud points, it forces the algorithm to focus on it during registration instead of other more relevant points belonging to the plants.

To address these issues, a custom ICP-SLAM algorithm was developed.

Custom ICP-SLAM Procedure

Considering the graphical scheme in Fig. 5.9, let us consider a set of point clouds PC_j $(j = 1 \dots J)$ acquired in the field, depicting each at least two plants of the orchard row. To all PC_j corresponds a reference system R_j. For each epoch $n = 1 \dots N$, the PCs are aligned in couples thanks to a transformation matrix T, thus halving the total number of j elements in the list. Hence, if at epoch $n = 1$ we align:

$$R_1^{(n=0)} \text{ and } R_2^{(n=0)} \rightarrow R_1^{(n=1)} \tag{5.1}$$

$$R_3^{(n=0)} \text{ and } R_4^{(n=0)} \rightarrow R_2^{(n=1)} \tag{5.2}$$

At epoch $n = 2$, we repeat this process obtaining:

$$R_1^{(n=2)} = R_1^{(n=0)} + R_2^{(n=0)} + R_3^{(n=0)} + R_4^{(n=0)} \tag{5.3}$$

which contains four merged point clouds.

By repeating this procedure, we can finally obtain only two macro point clouds corresponding to reference systems $R_1^{(n=N)}$ and $R_J^{(n=N)}$ (with $J = 2$, equal to the number of point clouds in the list at epoch N), each including half the original number of point clouds, that are merged to obtain PC_{final} (corresponding to a reference system R_{final}). In this way, we are able to:

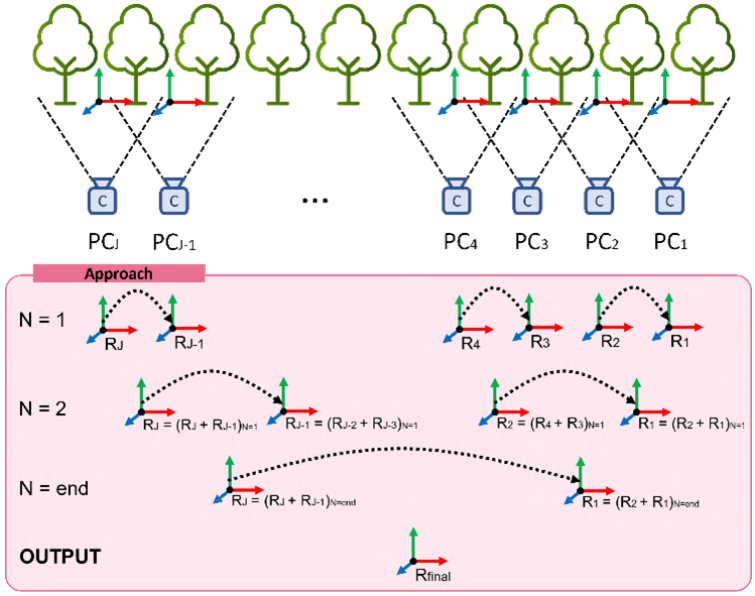


Figure 5.9: Scheme of the alignment procedure.

1. Remove the spatial drift that is observed between point clouds.

2. Ensure that the merging operation is always successful since it is applied on point clouds with a similar number of points.

To facilitate the reader, a scheme detailing the epoch processing is shown in Fig. 5.10 and the merging operation is shown in Fig. 5.11.

Processing Steps

1. **Initialization:** for each epoch n, the process starts taking as input a list of j point clouds PC_j, sorted according to their acquisition time (timestamp). The iteration counter j starts from $j = 2$, corresponding to the source point cloud PC_j. The epoch iterates over pairs of consecutive PC_j and PC_{j-1} that will be registered (source and target point clouds respectively). Please note that the procedure always considers couples of point clouds as

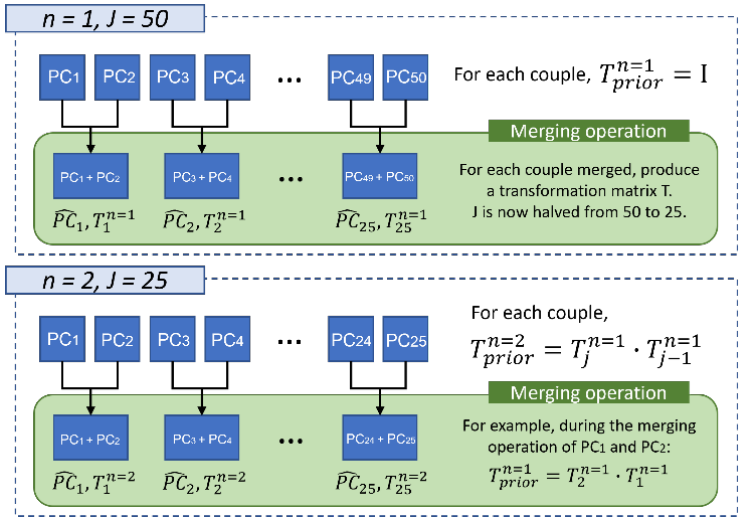


Figure 5.10: Scheme detailing the processing of the first and second epochs ($n = 1$ and $n = 2$). The consequent epochs are always processed in a similar fashion as epoch 2.

graphically depicted in Fig. 5.9 and Fig. 5.10.

2. **Coarse alignment:** to deal with the spatial drift, an initial coarse alignment is performed on the source point cloud PC_j. First, the current T_{prior}^{n} is calculated as the dot product between the transformation matrices produced from the alignment process of the previous epoch:

$$T_{prior}^{n} = T_j^{(n-1)} \cdot T_{j-1}^{(n-1)} \tag{5.4}$$

For the first epoch, $T_{prior}^{(n=1)}$ is initialized as the identity matrix. Then, the coarse alignment is performed on the source point cloud, producing:

$$S_t = PC_j \cdot T_{prior}^{n} \tag{5.5}$$

3. **ICP registration:** the ICP algorithm is applied to the transformed source point cloud S_t and the target point cloud PC_{j-1}, iteratively moving S_t onto

PC_{j-1}. The algorithm minimizes the distance between the corresponding points of the two point clouds and produces a transformation matrix T^n_{ICP}.

4. **Output generation:** the output of the merging operation is twofold:

$$T^n_j = T^n_{prior} \cdot T^n_{ICP} \tag{5.6}$$

$$\hat{PC}_j = S_t \cdot T^n_{ICP} + PC_{j-1} \tag{5.7}$$

5. **Exception for odd values of J:** if the total number of point clouds in the list is odd, the idea of merging in couples fails. In this situation, the procedure simply skips the last point cloud, which is instead passed to the next epoch without transformation. Referring to the example in Fig. 5.10, at epoch $n = 3$, PC_{25} is not processed. Hence, at epoch $n = 4$, the total number of point clouds will be:

$$J = \frac{24}{2} + 1 = 13 \tag{5.8}$$

Front and Back Merging

To further improve the 3D representation of the orchard and better visualize the plant details, we performed the merging operation on both the data of the front (obtaining a merged "front" point cloud) and on the data of the back of the same orchard (obtaining a merged "back" point cloud).

The two are then fused using the same merging procedure detailed in Section III.B to create a comprehensive reconstruction of the orchard, referred to as PC_{double_side}. It is worth noting that from both point clouds the ground was removed to avoid potential errors of the algorithm (since most of the points belong to the ground instead of the plants); hence, the merging is performed considering only the points belonging to the orchard.

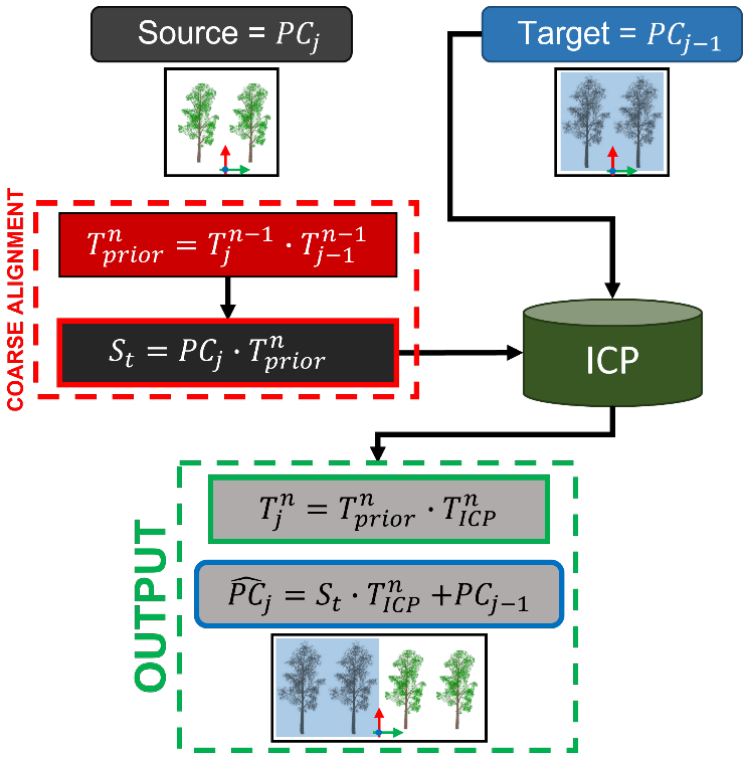


Figure 5.11: Scheme of the merging operation.

Figure 5.12 shows an example of a single point cloud and an example of the merged point cloud obtained at the end of the procedure.

5.4 Evaluation

5.4.1 Evaluation Dataset

After applying the methodology described in Section III to the point clouds acquired by Kinect Azure, we produced a total of six reconstructed point clouds PC_{double_side}. From the Viametris scanner, we obtained the same number of point clouds, denoted as $PC_{viametris}$. Finally, for validation purposes, we included in our benchmark the six PC_{single_view} point clouds corresponding to the front view.

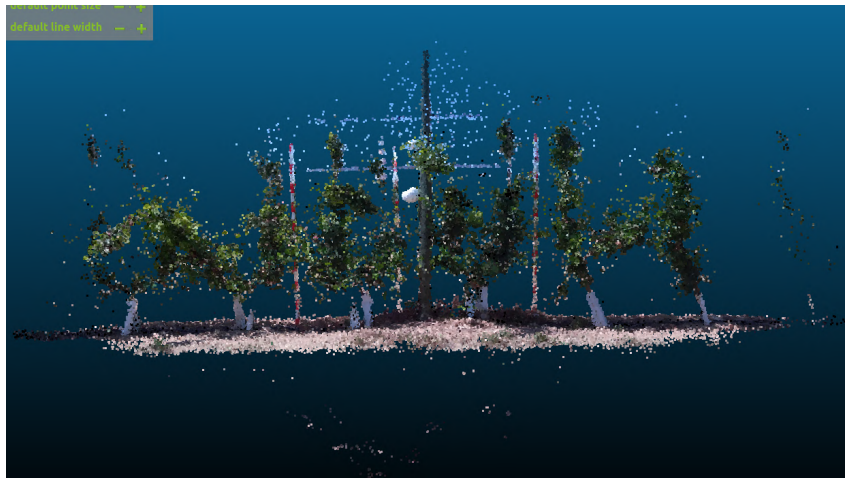
Since our aim is to validate the procedure using reference objects that simulate apples, we obtained a subsection of each reconstructed point cloud (denoted as $PC^{sub}_{double\_side}$, $PC^{sub}_{viametris}$, and $PC^{sub}_{single\_view}$ accordingly), corresponding to the portion of the row in which the experimental set-up detailed in Section II.A was mounted. The subsection area was manually selected considering the red and white poles as threshold references (Fig. 2.1). This ensures that the subsection corresponds to the same reference area for each point cloud.

To evaluate the quality of the alignment procedure, we analyzed the reconstruction of the reference foam spheres that can be seen in all the subsectioned point clouds $PC^{sub}_{double\_side}$, $PC^{sub}_{viametris}$, and $PC^{sub}_{single\_view}$. We utilized the six spheres (with radii 97 mm, 72 mm, 48.5 mm, 38.5 mm, 28 mm, and 24 mm, respectively) as the reference targets to evaluate the reconstruction quality. From each point cloud, points belonging to the six spheres were manually extracted so that their data was separated from the rest of the point cloud.

5.4.2 Evaluation Methodology

The evaluation metrics considered for benchmarking are:

(a) Example of a single point cloud acquired in the field.



(b) Example of the merged point cloud obtained at the end of the procedure, including both views of the orchard (front and back).

Figure 5.12: Comparison between a single point cloud and the final merged point cloud.

1. The radius of each sphere computed from the reconstructed point clouds and its associated uncertainty relative to the actual diameter of the sphere (ground-truth).

2. The sphericity of the reconstructed spheres.

3. The point density of the spheres.

To measure the radius, we first shifted the point cloud to the origin of the reference frame by calculating its center as the mean of all points and moving it to coordinates $(0, 0, 0)$. Then, a spherical volume V is calculated as the 95th percentile of all points in the sphere's point cloud. The center of V is the mean value of the sphere, and the radius is the 95th percentile of the distances between these points and the center. This percentile is chosen to minimize the impact of outliers and provide a robust estimate, ensuring that the radius measurement reflects the true size of the sphere by focusing on the bulk of the data.

The uncertainty estimation of the radius is based on the non-sphericity of V. We first compute the convex hull H of each sphere, defined by [17] as the smallest convex polyhedron that contains all the points in the set. Then, we subtract H from V. If this step produces a value of zero, then the fitting is perfect and there is no uncertainty. However, this condition never verifies; instead, we are left with a portion of the volume considered as the uncertainty on the volume computation, denoted as V_σ. Using the "Guide to the Expression of Uncertainty in Measurement (GUM)" [18], we can estimate the uncertainty on the radius estimation, σ, from V_σ. This value is then extended to the 95% confidence interval.

A second analysis is based on sphericity. According to Wadell's method [19], the sphericity (Ψ) of a point cloud representing a spherical object measures how closely an object's shape resembles a perfect sphere. It is the ratio of the surface area of a sphere, having the same volume as the object, to the actual surface area of the object. This can be expressed mathematically as:

$$\Psi = \frac{\pi^{1/3}(6V)^{2/3}}{A} \tag{5.9}$$

For our analysis, we first calculate the convex hull H of the point cloud to determine the volume (V) and surface area (A) of the object. We then compare these values to those of a theoretical sphere with the same volume to compute the sphericity Ψ. This method accurately reflects the object's deviation from a perfect spherical shape, providing a robust metric for assessing spherical representations in metrological applications.

Finally, we calculated the density ρ of the ideal sphere (expressed in points/m^3) to obtain insights about the reliability of the extracted data. The formulation to compute it is:

$$\rho = \frac{\text{number of points}}{\text{volume of the ideal sphere}} \tag{5.10}$$

5.4.3 Results

The difference between the ground-truth radius and the estimated one is calculated and displayed in Fig. 5.13, in which the corresponding uncertainty is displayed as an error bar. For the $PC^{sub}_{viametris}$ point clouds, we observe a relatively stable offset error across different radii, indicating consistent performance regardless of object size.

In contrast, the Kinect Azure (KA) point clouds $PC^{sub}_{double\_side}$ and $PC^{sub}_{single\_view}$ show larger offset errors in the case of smaller spheres. This error decreases as the sphere size increases, suggesting that the Kinect Azure struggles with smaller objects due to lower point cloud density and spatial resolution limitations. This significant systematic error highlights the need for more advanced radius estimation techniques. Potential methods to address this issue include improved calibration algorithms and machine learning approaches to enhance accuracy.

The sphericity of objects against their real radius is shown in Fig. 5.14. The Viametris system consistently performs well regardless of the size of the spherical targets. As for KA, a good sphericity higher than 85% is obtained for spheres of radius greater than 40 mm, while for smaller radii, the sphericity values drop to around 65-75%. These values are still acceptable for most applications despite

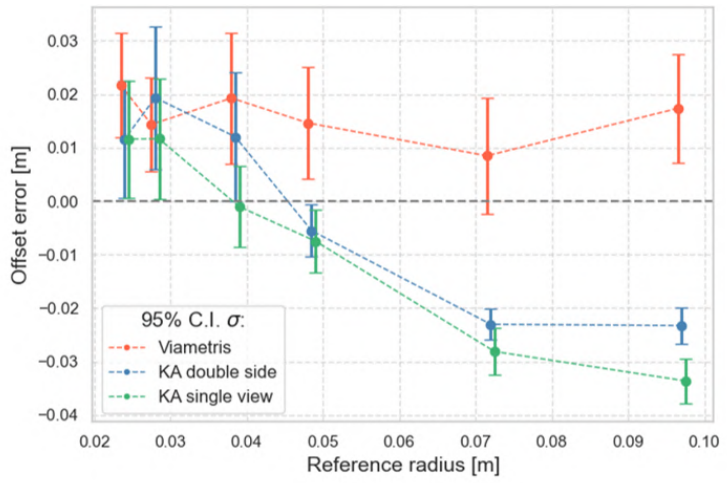


Figure 5.13: Difference between the ground-truth radius and the estimated one. Data points with error bars represent measurement uncertainty.

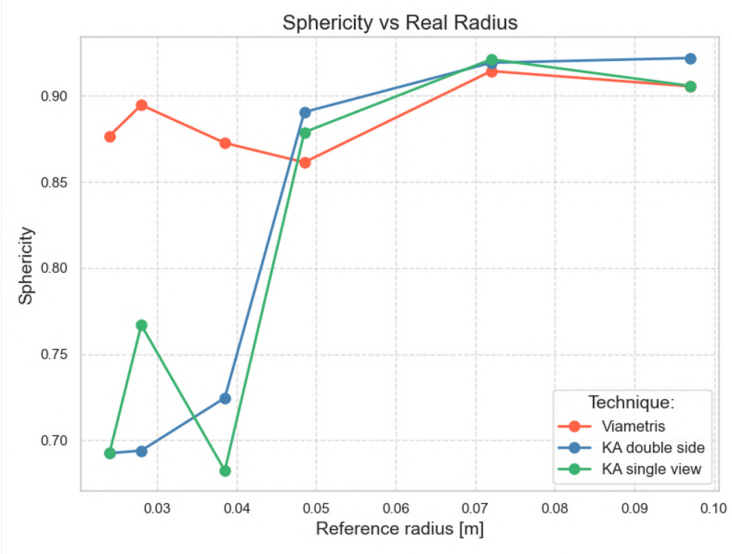


Figure 5.14: Sphericity values according to the ground-truth radius of each sphere.

the limitations of ToF technology in outdoor environments and in reconstructing small objects.

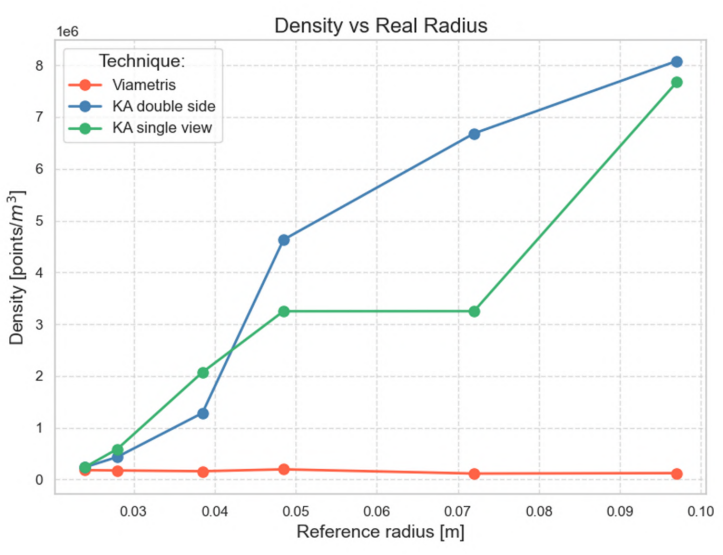


Figure 5.15: Density values plotted according to the ground-truth radius of the spheres. These values reflect the sensor performance and photonic technology characteristics.

The density of the spheres' point clouds against their ground-truth radius is illustrated in Fig. 5.15. The Viametris system has a point cloud density that is significantly lower compared to the other two; however, it is still capable of obtaining sphericity values higher than 85%. This is due to the laser scanner technology, which produces accurate point clouds without excessive density. In contrast, the KA shows increasing density with larger spheres. This demonstrates that the sensor captures more details for larger objects.

Finally, by taking into account all results, it is evident that $PC^{sub}_{double\_side}$ point clouds have no significant advantages compared to $PC^{sub}_{single\_view}$ point clouds, indicating that the latter should be the preferable choice.

5.4.4 Conclusions

This study demonstrated the feasibility of using low-cost RGB-D cameras through SLAM for generating 3D reconstructions of an apple orchard. We developed a custom methodology to fuse single-shot point clouds into a denser one that represents the full orchard (which is several meters long). The experimental method was validated considering six target spherical objects of increasing radius, ranging from 24 mm to 97 mm. The spheres were reconstructed, and their radius was computed with the associated uncertainty. We also obtained the sphericity measure and point density. These three metrics were used to compare the results of the Viametris system and two reconstruction methods for data obtained from Kinect Azure: single view and double side, the latter including both the front and back of the orchard.

The project primarily focused on evaluating the minimum resolution of the RGB-D camera, with the objective of distinguishing the smallest possible objects and determining its technological limits. The broader goal was to create a reliable 3D reconstruction of the orchard, including both small plant organs and macroscopic features such as trunks and canopies. This reconstruction can be interpreted by algorithms or agronomists to extract useful data according to specific needs. In line with this objective, we chose reference objects that were compatible with the dimensions of small organs like fruits, buds, and branches.

Our results showed that while the Kinect Azure provided comparable reconstruction quality for objects with a radius larger than 45 mm, it struggled with smaller ones, revealing the limitations of Time-of-Flight (ToF) technology in contrast to laser scanning. Additionally, there were no significant advantages to adopting the double-sided point cloud over the single-view reconstruction.

Ongoing analyses are being performed to compare the performance of the LiDAR system with the Kinect Azure, which will be detailed in a future publication. Preliminary static acquisitions were carried out over an entire day to identify optimal lighting conditions for the sensors, using multiple lux meters to record brightness levels. These findings will be published in a subsequent work. As the sensor uses global shutter technology, dynamic acquisition characteristics

only affect the transformation between one point cloud and another. Disturbances such as vibrations and speed have been addressed in extended work through the use of a high-accuracy Xsens accelerometer. While other factors, such as environmental conditions, may be of interest, our current work concentrated on sensor comparison under the same operational conditions.

Future improvements include:

- Acquisitions at twilight to avoid the harsh lighting of the day.
- Capturing point clouds at closer distances.
- Implementing a closed-loop SLAM fusing methodology to enhance the accuracy of the reconstructed fused point cloud.

A key future step is the autonomous recognition, detection, and segmentation of objects using Convolutional Neural Networks (CNN), allowing for the precise extraction of point cloud portions.

In summary, integrating low-cost sensors with advanced data processing techniques holds great potential for detailed and accurate 3D reconstructions in agriculture, though further refinements are needed.

5.5 Future Directions: Modified Alignment for Large Point Clouds

Algorithmic Challenge: Decreasing Relative Overlap

As the size of point clouds grows over epochs, the percentage overlap decreases despite constant absolute overlap. This limitation impacts the effectiveness of the Iterative Closest Point (ICP) algorithm, which relies on sufficient overlapping regions for accurate registration.

Proposed Solution: Partial Segmentation

To address this challenge, a segmentation technique was implemented to restrict the alignment process to a specific region of interest near the centroid of overlap. This approach enhances stability and significantly reduces computational costs, particularly in later epochs where point clouds can span tens of meters but the overlapping area remains only slightly larger than the camera's field of view.

Implementation

Centroid Computation: The centroids of the source and target clouds are computed as:

$$C_s = \frac{1}{n}\sum_{i=1}^{n} P_{s,i}, \quad C_t = \frac{1}{m}\sum_{i=1}^{m} P_{t,i}$$

where $P_{s,i}$ and $P_{t,i}$ represent the points in the source and target clouds, respectively.

Defining the Area of Interest: A spherical region with radius r is centered at the midpoint of the centroids:

$$M = \frac{C_s + C_t}{2}$$

Radius Filtering: Points within r of the midpoint are retained using:

$$\|P - M\| \leq r$$

Dynamic Radius Adjustment: The radius r is dynamically estimated based on the average row height, calculated as the 95th percentile of plant height (h_{95}):

$$r = \frac{h_{95}}{2}$$

Here, h_{95} represents the height of the tallest plants within the 95th percentile of the point cloud, ensuring that the region of interest encompasses the relevant

part of the orchard row.

Results and Observations

This method improves ICP stability and efficiency by significantly reducing the number of points processed. The reduction in point cloud size is particularly impactful in later epochs, where point clouds can extend over tens of meters, while the overlapping region remains limited to the field of view of the camera. This optimization greatly decreases processing time, enhancing the overall performance of the alignment process.

Conclusion

This approach dynamically adapts the alignment process for large point clouds, improving robustness and computational efficiency in real-time applications. By estimating the region of interest based on the average row height, the method ensures that relevant points are retained while minimizing computational overhead.

Bibliography

[1] Eduard Gregorio López and José Antonio Martínez Casasnovas. Low-cost orchard monitoring systems for precision agriculture based on photonic sensors (digifruit). Research project funded by the Ministerio de Ciencia e Innovación (TED2021-131871B-I00), 2022. Universitat de Lleida, Spain. https://www.grap.udl.cat/en/research/research-projects/digifruit/.

[2] E. Gregorio and J. Llorens. *Sensing crop geometry and structure*. Springer International Publishing, Cham, Switzerland, 2021.

[3] S. Jay, G. Rabatel, X. Hadoux, D. Moura, and N. Gorretta. In-field crop row phenotyping from 3d modeling performed using structure from motion. *Computers and Electronics in Agriculture*, 110:70–77, 2015.

[4] J. Martínez-Guanter, Á. Ribeiro, G.G. Peteinatos, M. Pérez-Ruiz, R. Gerhards, and J.M. et al. Bengochea-Guevara. Low-cost three-dimensional modelling of crop plants. *Sensors*, 19(13):2883, 2019.

[5] M. Herrero-Huerta, D. González-Aguilera, P. Rodriguez-Gonzalvez, and D. Hernández-López. Vineyard yield estimation by automatic 3d bunch modelling in field conditions. *Computers and Electronics in Agriculture*, 110:17–26, 2015.

[6] J. Gené-Mola, R. Sanz-Cortiella, J.R. Rosell-Polo, A. Escolà, and E. Gregorio. In-field apple size estimation using photogrammetry-derived 3d point clouds: Comparison of 4 different methods considering fruit occlusions. *Computers and Electronics in Agriculture*, 188:106343, 2021.

[7] A. Escolà, J.A. Martínez-Casasnovas, J. Rufat, J. Arnó, A. Arbonés, and F. et al. Sebé. Mobile terrestrial laser scanner applications in precision fruticulture/horticulture and tools to extract information from canopy point clouds. *Precision Agriculture*, 18:111–132, 2017.

[8] R. Sanz, J. Llorens, A. Escolà, J. Arnó, S. Planas, and C. et al. Román. Lidar and non-lidar-based canopy parameters to estimate the leaf area in fruit trees and vineyard. *Agricultural and Forest Meteorology*, 260-261:229–239, 2018.

[9] J. Gené-Mola, E. Gregorio, F. Auat Cheein, J. Guevara, J. Llorens, and R. et al. Sanz-Cortiella. Fruit detection, yield prediction and canopy geometric characterization using lidar with forced air flow. *Computers and Electronics in Agriculture*, 168:105121, 2020.

[10] J.R. Rosell-Polo, F. Aauat Cheein, E. Gregorio, D. Andújar, L. Puigdomènech, and J. et al. Masip. Advances in structured light sensors applications in precision agriculture and livestock farming. *Advances in Agronomy*, 133:71-112, 2015.

[11] J.R. Rosell-Polo, E. Gregorio, J. Gené, J. Llorens, X. Torrent, and J. et al. Arnó. Kinect v2 sensor-based mobile terrestrial laser scanner for agricultural outdoor applications. *IEEE/ASME Transactions on Mechatronics*, 22(6):2420-2427, 2017.

[12] J. Bengochea-Guevara, D. Andújar, F. Sanchez-Sardana, K. Cantuña, and A. Ribeiro. A low-cost approach to automatically obtain accurate 3d models of woody crops. *Sensors*, 18(1):30, 2017.

[13] A. Milella, R. Marani, A. Petitti, and G. Reina. In-field high throughput grapevine phenotyping with a consumer-grade depth camera. *Computers and Electronics in Agriculture*, 156:293–306, 2019.

[14] Bernardo Lanza, Ricardo Sanz-Cortiella, Alexandre Escolà-Agustí, Marc Felip-Pomés, Simone Pasinetti, Jaume Lordan-Sanahuja, Jordi Gené-Mola, José M. Plata-Moreno, Eduard Gregorio-López, Cristina Nuzzi, and José A. Martínez-Casasnovas. Metrological assessment of rgb-d cameras for 3d orchard reconstruction. *Metrology for Agriculture and Forestry*, 2024.

[15] Microsoft. Azure kinect dk hardware specifications, 2022. Available online: https://learn.microsoft.com/en-us/azure/kinect-dk/hardware-specification [Accessed 21 June 2024].

[16] S. Pasinetti, C. Nuzzi, A. Luchetti, M. Zanetti, M. Lancini, and M. De Cecco. Experimental procedure for the metrological characterization of time-of-flight cameras for human body 3d measurements. *Sensors*, 23(1):538, 2023.

[17] J. G. Hocking and G. S. Young. *Topology*. Addison-Wesley Publishing Co. Inc., 1961.

[18] Joint Committee for Guides in Metrology. Guide to the expression of uncertainty in measurement — part 6: Developing and using measurement models. 2020.

[19] H. Wadell. Volume, shape, and roundness of quartz particles. *The Journal of Geology*, 43(3):250–280, 1935.

Chapter 6

Conclusions

This dissertation has focused on the development and validation of optical measurement methodologies for in-field assessment of plant health and agricultural yield, integrating photonic sensors, AI-based data processing, and rigorous metrological validation. The core objective has been not only to measure relevant agronomic parameters but also to evaluate the measurement uncertainty associated with each sensor–algorithm setup, thereby ensuring applicability in real-world conditions. Every proposed methodology has been assessed in accordance with the principles of the *Guide to the Expression of Uncertainty in Measurement* (GUM), using quantitative estimations of measurement uncertainty derived from experimental field trials. These trials, conducted both in controlled laboratory settings and open-field environments—including vineyards at MASI Amarone (VR) and experimental farms in Spain—provided valuable reference data for performance evaluation under operational conditions.

The vineyard monitoring component involved a detailed **uncertainty quantification** for wood volume estimation, distinguishing between **systematic errors**, related to sensor calibration and model assumptions, and **random errors**, arising from measurement noise and environmental variability. This was addressed through a **cylindrical decomposition technique**, where vine shoots were segmented into discrete sub-elements, approximating their volume as a sum

of locally fitted cylinders. This method minimizes model bias by preserving local geometric variations and allows for a structured evaluation of measurement uncertainty, expressed in terms of both **repeatability** (variation in repeated measurements under identical conditions) and **reproducibility** (variation under different operational scenarios).

The bud detection framework was designed with an **anti-background optical strategy**, leveraging a controlled depth of field to isolate the target from background clutter, thereby reducing false positives without relying solely on computational segmentation. The effectiveness of this approach was assessed through an analysis of **detection uncertainty**, measured as a function of optical resolution, lighting conditions, and bud occlusion rate.

For depth estimation using monocular RGB sensors, the approach exploits the **relative motion of the acquisition platform** (e.g., a tractor) to infer depth from perspective-induced disparities. The uncertainty of this method was systematically characterized with respect to **camera motion dynamics**, **image resolution**, and **scene structure**. Experimental validation demonstrated that, within controlled conditions, the method achieves **depth estimation uncertainties comparable to dedicated stereo systems** within a constrained operating range.

In large-scale 3D reconstructions, a **non-incremental SLAM approach** was developed to mitigate cumulative drift errors, which are typical in sequential pose estimation. To ensure metrological consistency, **reference spheres** of known dimensions were introduced into the environment, allowing direct quantification of reconstruction uncertainty. The uncertainty associated with these reference objects was analyzed as a function of **sensor noise**, **point cloud density**, and **sphere dimension**, enabling an experimental evaluation of the technique's effectiveness in high-resolution agricultural mapping.

A key contribution of this research lies in the explicit treatment of measurement uncertainty, which defines the limits of applicability for each methodology. By quantifying how environmental conditions, sensor characteristics, and algorithmic parameters affect measurement accuracy and reliability, this work provides practical guidelines for deploying these techniques in precision agricul-

ture. Future research will focus on refining uncertainty propagation models, extending field validation to a broader range of agronomic conditions, further optimizing real-time processing for embedded implementations, and integrating additional imaging modalities—such as multispectral and thermal sensors—for more comprehensive crop health evaluation. The automation of data acquisition via autonomous platforms will also be explored, thereby enabling continuous and scalable monitoring in operational settings.

By adhering to metrological principles and validating all proposed methods through experimental uncertainty analysis, this dissertation establishes a robust framework for precision agriculture, bridging the gap between theoretical developments and practical, real-world deployment.