School of Mechanical and Manufacturing Engineering
Faculty of Engineering
UNSW Sydney

BY

David Nie

Optimising Comprehensive Plant Data Collection Using a Mobile Manipulator for Holistic Agricultural Insights

Thesis submitted as a requirement for the degree of Bachelor of Engineering in Mechanical Engineering

Submitted: 07/08/2025	Student zID: z5303303
Supervisor: Leo Wu (UNSW)	

ORIGINALITY STATEMENT

'I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.'

Signed: ____________________

Date: Aug 7th, 2025

Abstract

This study demonstrates a proof-of-concept for vision-based fruit harvesting using robotic systems, validating that while technically feasible, full autonomy remains inefficient compared to semi-autonomous approaches. The work connects to broader agricultural robotics challenges outlined in [24], showing how unstructured environments and task complexity limit performance - our system achieved 17-20s cycles for horizontal tasks but required 75s for vertical harvesting due to motion planning constraints. The results support the hypothesis that human-robot collaboration outperforms full autonomy in adaptability, while also confirming literature critiques on productivity gaps. Future integration of advanced planners and task-specific end-effectors could bridge this efficiency gap, though fundamental redesigns may be needed for commercial viability.

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my supervisor, Dr. Leo Wu, for his invaluable guidance, encouragement, and support throughout this project. I would also like to thank Jean-Henri Odendaal for his consistent assistance and technical help in the laboratory.

I am also thankful to the entire MoMa team: Taha Inam, Weichen Tie, and Chun Fong Wong. This work would not have been possible without the collaborative spirit, mutual support, and dedication we shared throughout the journey.

Contents

List of Figures

Nomenclature

(The nomenclature must respect the following order: latin symbols, Greek symbols, acronyms. Within each category of symbols, follow alphabetic order. An example is provided below)

AMR	Autonomous Mobile Robot
DOF	Degree of Freedom
FOV	Field Of View
HO	Human Operator
HRC	Human Robot Collaboration
HRS	Human Robot System
MAE	Mean Absolute Error
MMW	Millimeter-wave
$MoMa$	Mobile Manipulator
MS	Multispectral
MSC	Multiple Scatter Correction
PA	Precision agriculture
$PLSR$	Partial Least Squares Regression
RWC	Relative Water Content
$SLAM$	Simultaneous Localization and Mapping
SNV	Standard Normal Variate
SWC	Soil water content
TH	Thermal
TOF	Time Of Flight
UAV	Unmanned Aerial Vehicles
VWC	Volumetric water content

1 Introduction

Agricultural robotics face a fundamental tension between autonomy and practicality - while sensors like mmWave [35] and multispectral systems [20] enable sophisticated crop monitoring, their translation to field operations remains constrained by environmental variability and economic realities. This disconnect mirrors broader findings in [25], where data collection capabilities outpaced actionable implementations. Our study tests two hypotheses emerging from this literature: first, that human-robot systems (HRS) provide superior robustness in unstructured environments compared to full autonomy, and second, that manual spatial optimisation enhances harvesting efficiency.

Prior work establishes critical benchmarks - mmWave systems like Leafeon [8] achieve 55.7% error reduction in leaf moisture detection, while multispectral arrays deliver 90.1% disease classification accuracy [20]. However, as [3] notes, such technical capabilities often fail to translate to field reliability without human oversight. The 80/20 rule proposed by [29] formalises this observation, suggesting machines should handle routine tasks while humans manage exceptions.

Our experimental framework (Figure 1) applies these principles to fruit harvesting using vision systems instead of the spectral sensors discussed in literature. This shift acknowledges practical constraints where spectral analysis performs well in laboratory settings, such as [31]'s 95% seed vigour classification, but real world harvesting requires robust geometric perception and manipulation capabilities.

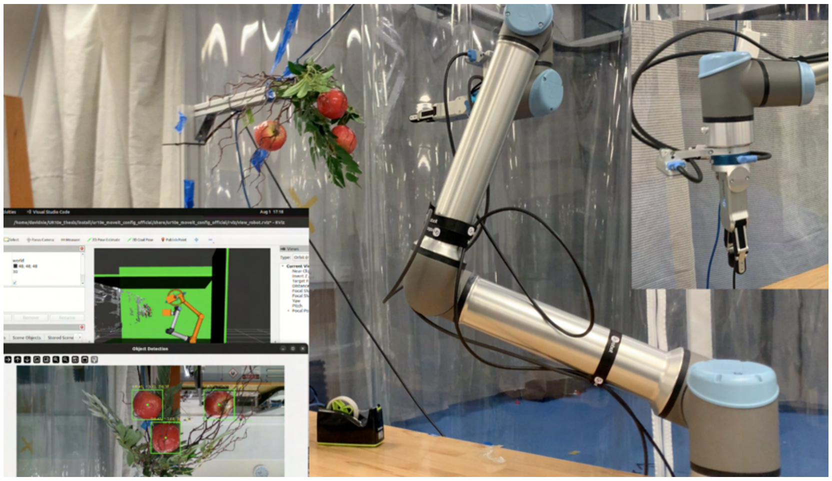


Figure 1: Experimental setup for vision-based fruit harvesting

This introduction leads to three research questions:

1. Can vision-based harvesting meet basic functionality requirements?

2. How does efficiency compare between autonomous and semi-autonomous modes?

3. What architectural improvements could close the performance gap? The subsequent sections address these through system implementation and controlled experiments.

2 Literature Review

2.1 Background

Global population growth is projected to reach 9.8 billion by 2050, with half of this increase concentrated in nine countries. This rise intensifies demand on agriculture to double food production, particularly of high-quality, pesticide-free food. Simultaneously, urbanization is reducing rural and arable land, with 68% of people expected to live in cities by 2050, challenging farmers to produce more in smaller areas [10].

As a response, [23] stated that agricultural productivity has grown significantly with automation and mechanization, which has enhanced crop yield, quality, and efficiency while reducing labor needs and environmental impact. Techniques like precision seeding, fertigation, and in-season nitrogen management allow farmers to maximize yield and uniformity, minimizing input costs and pollution [23]. Robotics, including autonomous tractors, have been introduced to help manage labor shortages, as they can reduce fuel usage and environmental impacts [23][11][3]. However, agriculture's unstructured environments present challenges, demanding adaptable robots to handle variables like plant shape, size, and texture, as well as changing conditions in soil and weather[23]. This is demonstrated in Fig.2, which shows how the direction of sunlight and varying illumination levels affect the visibility of pepper rows in a greenhouse. Moreover, development requires advanced algorithms to navigate unstructured, dynamic conditions and adapt to diverse crops and tasks [3]. Seasonality reduces robot utilization, hindering cost-effectiveness compared to manufacturing. Partial automation often succeeds in simpler tasks, but achieving full autonomy for complex tasks remains challenging due to high variability and production inefficiencies [24]. This complexity makes designing and deploying agricultural robots challenging and expensive, with much of the work—like harvesting, sorting, and packaging—still requiring human labor.

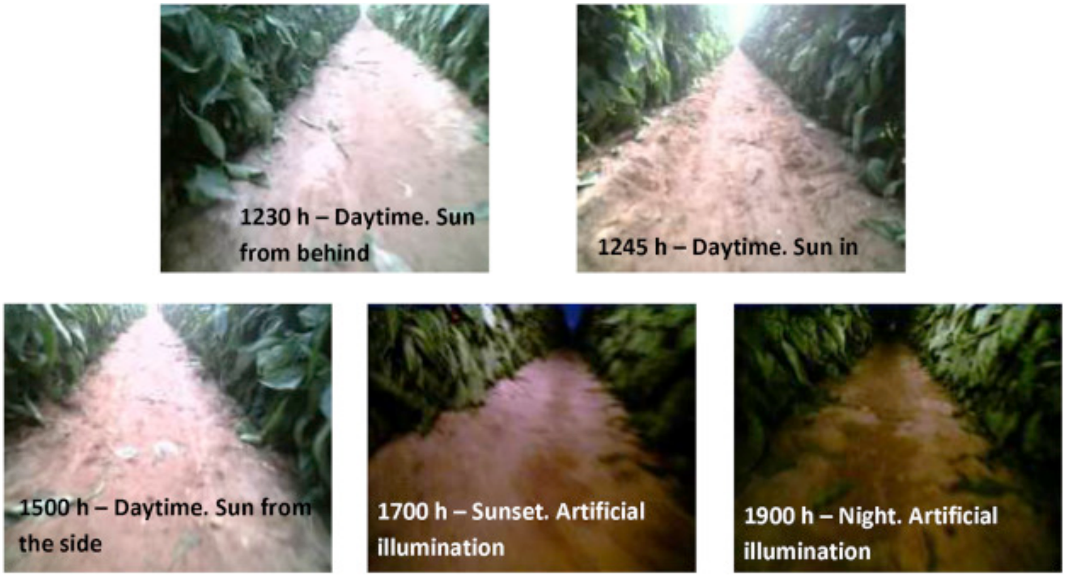

1230 h – Daytime. Sun from behind
1245 h – Daytime. Sun in
1500 h – Daytime. Sun from the side
1700 h – Sunset. Artificial illumination
1900 h – Night. Artificial illumination

Figure 2: Images of a pepper row in a greenhouse were captured from a robotic platform at five different times throughout the day [12]

[23] suggested that improving agricultural robotics requires advances in four key areas: locomotion systems, sensors, computer vision algorithms, and IoT-based smart agriculture. Current systems often lack versatility, robust navigation, and adaptability for different crops. Enhanced locomotion and sensor technology could enable robots to handle diverse terrains and tasks more effectively. Advances in computer vision and IoT would allow better crop monitoring, weed identification, and overall farm management.

However, the proposed improvements are easier said than done. [25] points out that the challenge in precision agriculture lies in bridging the gap between increasing data availability and actionable insights. While better agronomic understanding requires more data, practical applications have stagnated. To advance, smart sensor systems must incorporate multi-causal decision-making, transforming raw data into knowledge. Robotics and automation must integrate this data seamlessly for real-time decision-making, helping farmers optimize processes while considering local and environmental factors. The development of these intelligent systems aims to automate, analyze, and recommend actions efficiently, yet still requires continuous refinement for widespread practical use [25][33].

[25] categories agricultural robotics into three applications: (i) weed control and targeted spraying, (ii) field scouting and data collection, and (iii) automated harvesting. [25] states that Data collecting robots often incorporates extensive use of advanced sensors for navigation control, manipulator control, obstacle avoidance, and three-dimensional environment reconstructions. They are also expected to be flexible, multipurpose, and affordable to be viable for commercial use.

In terms of the use of sensors, [6] states that portable, fixed, and wireless sensors are used to track soil quality, nutrient levels, weather, and crop health, providing essential data for precision agriculture. [22] points out that there are several needs for future research in precision farming:

1. Enhanced chemometric methods for decomposing spectral data, leveraging hyperspectral sensors' fine spatial and spectral resolutions.

2. Sensors for nutrient deficiencies that eliminate the need for reference strips, enabling more direct, on-the-go assessments.

3. New spectral indices for monitoring multiple crop attributes (e.g., biomass, LAI) and stressors (e.g., water, pests) simultaneously.

4. Data integration from historical archives with current high-resolution data for more accurate and responsive decision-making in agriculture.

While [25] states that it is not realistic to expect an entirely automated farming system in the future, [29] proposed the 80/20 rule as a mitigation method. The paper claims that approximately 80% of a task is easy to perform, while 20% is difficult. In the context of semi-autonomous systems, this means that machines can handle the simpler, more repetitive parts of a task, performing them faster or more accurately than humans. The human operator can focus on the more complex or difficult 20%, such as dealing with unexpected situations. This type of automation, known as "operator assist," can improve efficiency and safety while still leveraging human expertise where it's most needed.

The concept of so-called Human Robot System (HRS) is backed by [9], which addressed key challenges like navigating fields, detecting produce, and harvesting, showing significant improvements over fully autonomous systems. Similarly, other research has demonstrated how HRS can outperform manual detection and autonomous systems in tasks like melon detection, with improved efficiency and reduced detection times [3].

As a result, there is a growing trend in agricultural robotics to transition from merely supervising robots to actively interacting with them, a development that has streamlined processes and enhanced system adaptability [3]. These advancements enable dynamic switching between collaboration levels based on task or environmental changes, ensuring optimal performance.

2.2 Applications of MmWave Sensors in Agriculture

Millimeter wave (mmWave) technology, which operates between 30 and 300 GHz, has been standardized in wireless communication protocols like IEEE 802.11ad, 802.11aj, and 802.11ay, and is also part of the 5G network [30]. This spectrum offers significantly higher bandwidth (10x more than 5 GHz WiFi) and provides faster data rates and greater communication density, enabling more simultaneous connections in confined areas. Additionally, mmWave's narrow beams and high frequencies make it ideal for applications requiring high-resolution sensing and object detection, as it can distinguish between different materials based on their interactions with the waves [30][16].

However, mmWave technology faces challenges such as signal attenuation over long distances, difficulties penetrating solid materials like concrete, and significant performance degradation in heavy rain [30][16].

Despite the challenges, mmWave sensors are increasing being used by researchers in the field of agriculture, due to its strong anti-interference and perception capabilities in complex and volatile environments [35]. [26] introduced the PELICAN radar shown in Fig.3, a lightweight (5 kg), low-power K-band radar (24 GHz) with a fan-beam antenna. This radar, designed for mobile robotics, can construct 2D representations of the surrounding environments and is adaptable for various platforms.

The simulated radar image is shown in Fig.4. Mark A indicates the dead zone introduced by the radar height and the antenna elevation aperture. Shadowing effect can be observed behind the buildings (marks B and C), behind the walls (mark D) and behind the hills (mark E). P1 and P2 are echoes from poles (considered as point targets) [26].

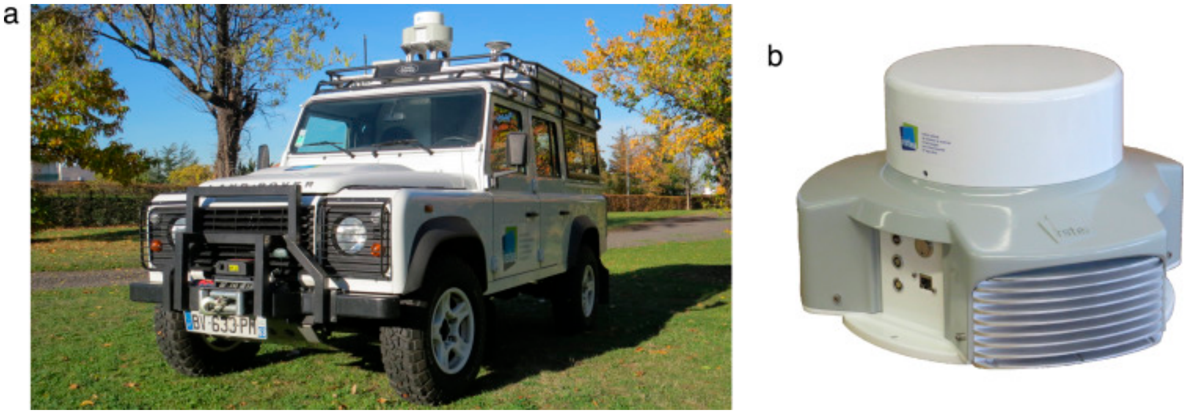


Figure 3: PELICAN radar (a) positioned on an experimental vehicle, and (b) detailed view [26].

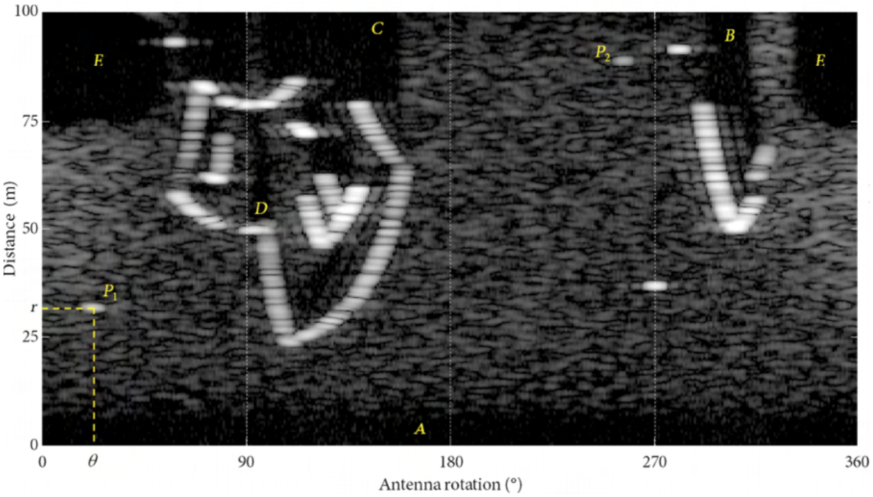


Figure 4: Simulated 2D radar image, in polar coordinates [26]

Furthermore, [21] proposes a method to estimate grape feature parameters using point cloud data, which involves segmenting the grape point cloud with filtering and region-growing algorithms, and registering it using an improved iterative closest point (ICP) algorithm. The grape bunch surface is then reconstructed using the Poisson algorithm, shown in Fig.5.

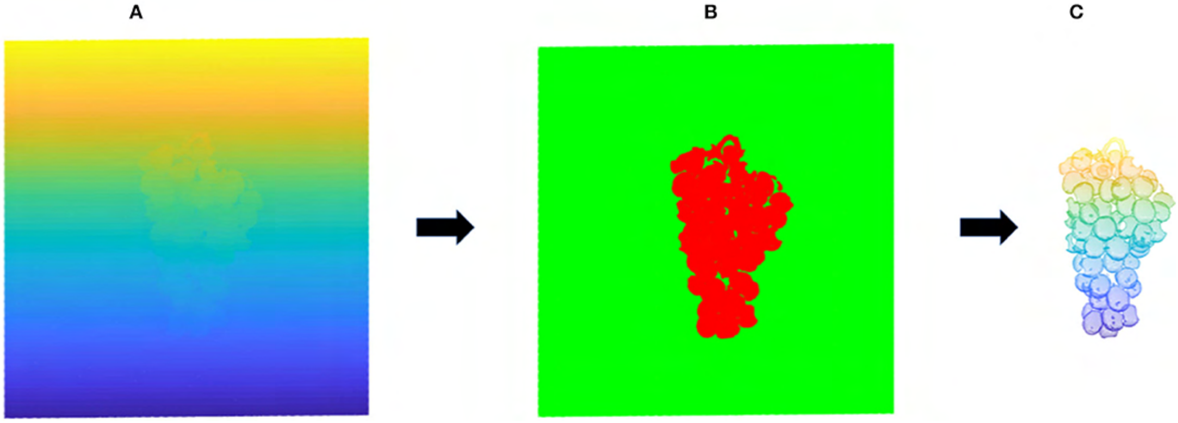


Figure 5: Point cloud segmentation map. (A) Represents the original point cloud image; (B) represents the clipping of outliers; (C) represents the grape point cloud extraction [21]

Moreover, mmWave sensors are used as a non-destructive method for analyzing sugar content in fruit. As fruit ripens, its sugar content increases, while starch content decreases, lowering its relative permittivity. Measuring the sugar content in whole fruit can enhance harvest quality[28]. [35] presents a method to extract three-dimensional canopy information from potted plants using mmWave radar, aiming to enhance precision in pesticide application, shown in Fig.6.

Recently, Leafeon, a plant sensing system that uses mmWave radar to measure leaf water content non-invasively, was introduced [8]. Leafeon addresses the impact of leaf structure and surface roughness on measurement accuracy by employing electronic beam steering to capture multiple perspectives of a

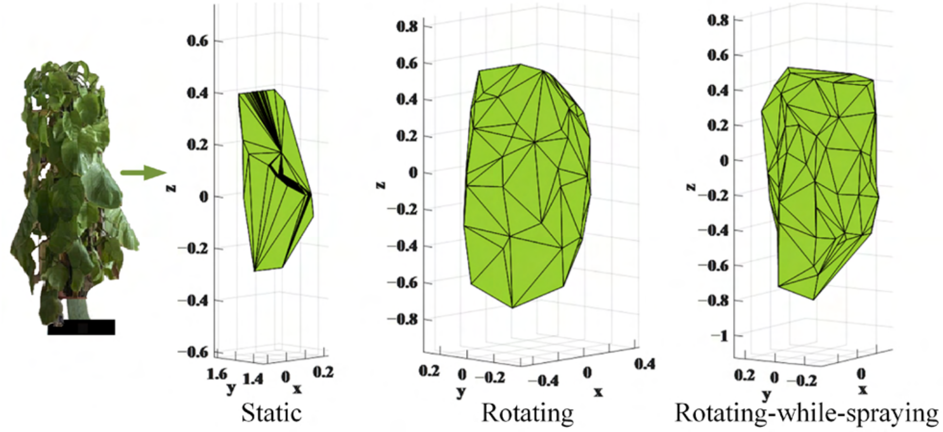
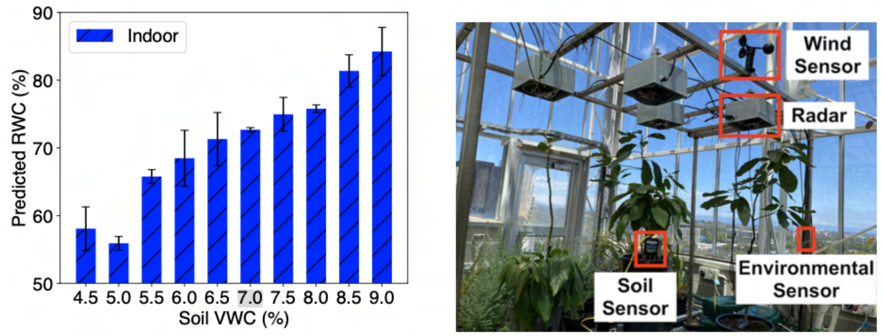


Figure 6: Potted plant point cloud reconstruction diagram

leaf. These perspectives are analyzed by a custom deep neural network that detects unique reflection patterns, enhancing estimation accuracy. Implemented with a commercial mmWave radar, Leafeon was tested on various leaf types and achieved a Mean Absolute Error (MAE) of 3.17% for avocado leaves, reducing error by up to 55.7% compared to existing methods [17]. Field experiments in both indoor and glasshouse settings showed that Leafeon's predicted leaf water content levels correlated closely with drought conditions, demonstrating its potential for real-time plant health monitoring in diverse agricultural environments.



(a) Predicted RWC vs Measured Soil VWC (indoor)

(b) Glasshouse farm experimental setup

Figure 7: Relationship between the predicted leaf RWC and the measured soil VWC [8]

In Leafeon's outdoor experiment setup, four mmWave radars, each focused on a single leaf, took hourly measurements to assess changes in leaf water content under natural sunlight [8]. However, using only four leaves to estimate a plant's overall water content may not provide a comprehensive measurement for larger plants. Additionally, the complex and stationary setup presents challenges in terms of practicality for broader agricultural applications.

From a broader perspective, more research are conducted towards object classification using mmWave radar which involves calculating the target's distance and reflection coefficient from the received attenuated signal [19] [32]. The relative permittivity can then be determined, which helps identify different materials. However, a limitation of this approach is the challenge of distinguishing materials with similar relative permittivity.

[32] introduced mSense, a mobile material sensing system that uses a single millimeter-wave (mmWave) radio to identify materials by analyzing reflected RF signals, rather than those penetrating the target. This approach avoids the need for multiple transceivers or sensor attachment to the target. Shown in Fig.8, mSense leverages a novel material reflection feature to quantify a material's reflectivity and employs techniques to ensure accurate material identification despite challenges such as device mobility, hardware limitations, and environmental interference. Experiments with commercial mmWave chipsets achieved an average accuracy of 93% in identifying five materials (aluminum, ceramic, plastic, wood, and water), with accuracy remaining around 90% in mobile scenarios. The experiment setup is shown in Fig.9.

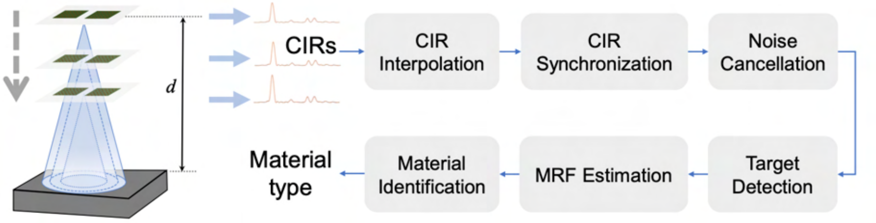


Figure 8: Overview of the mSense workflow

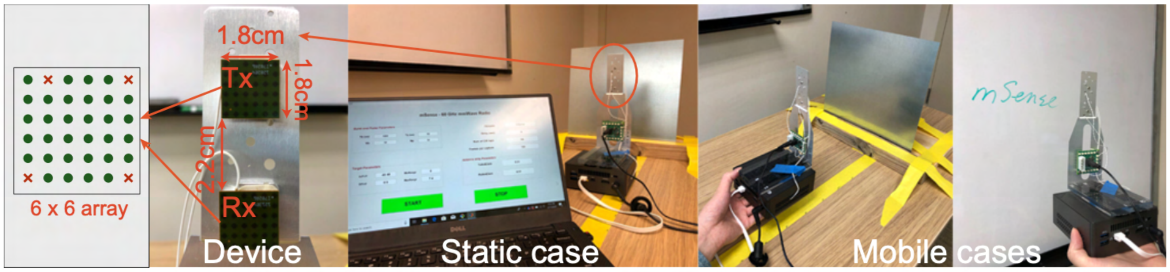


Figure 9: mSense testbed and example usage scenarios

While the paper emphasizes its focus on material identification for everyday objects made from common materials such as metals, plastics, and wood, rather than highly precise, critical applications [32]. mSense cannot detect targets that are blocked or hidden by other objects, and struggles with more complex shapes, such as soft and flexible materials like clothing.

[19] introduces FG-LiquID, a robust system for fine-grained liquid identification using mmWave radar. Unlike previous systems that focus on coarse material identification, FG-LiquID targets the small but informative region of the mmWave spectrum to extract discriminative features of liquids. The system uses a novel neural network to analyze signal patterns across multiple antennas and calibrates signals to eliminate location interference from minor container displacement or rotation, shown in Fig.10. FG-LiquID demonstrated high accuracy, distinguishing 30 different liquids with an average

accuracy of 97%, even identifying similar liquids like those with only a 1% difference in alcohol content. The system proves effective across five different scenarios, showcasing its potential for applications like detecting alcohol content, spoiled milk, and water contamination.

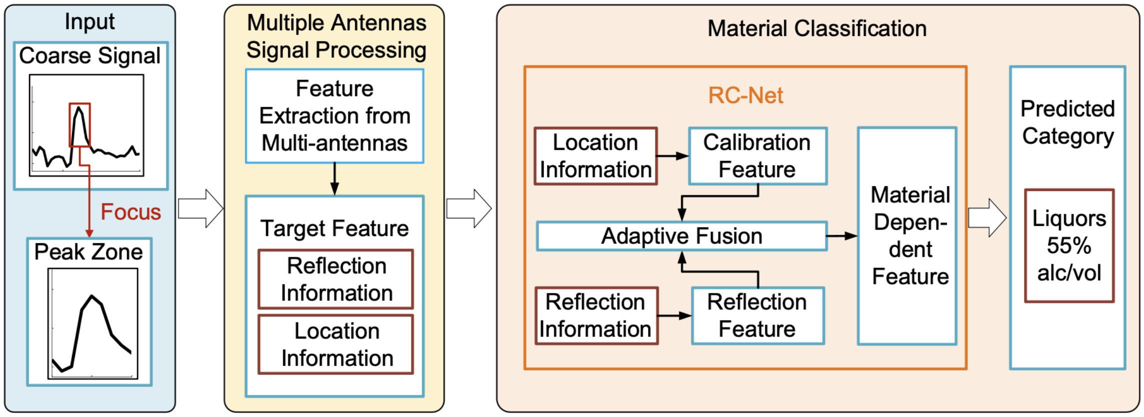


Figure 10: System workflow of FG-LiquID. [19]

However, FG-LiquID struggles with identifying objects placed randomly due to the lack of training data for such scenarios. On a similar note, when containers move, the Doppler effect causes significant changes in reflected signals, which may interfere with liquid identification [19]. FG-LiquID also requires containers to have a regular, symmetrical shape for robust identification, and introducing a new container requires additional data collection [19]. Lastly, the system performs poorly with metal containers due to signal attenuation.

2.3 Applications of Multispectral Sensors in Agriculture

Multispectral technology, which combines spatial and spectral imaging [15], has significantly advanced since its inception in the 1960s, initially used in remote sensing and military applications [34]. The development of semiconductor photodetectors accelerated its growth, expanding its applications to agriculture, environmental science, and food engineering.

Spectral imaging generates three-dimensional data sets by capturing multiple images of the same object at different wavelengths [15]. In agriculture, multispectral imaging is widely used for grain yield prediction [37], pest and disease detection [20] [18], water content detection [2] [4].

Illustrated in Fig.11, [20] explores the use of unmanned aerial vehicles (UAVs) for detecting eucalyptus leaf diseases using high-resolution multispectral imagery. The research identified key spectral bands and vegetation indices, such as green, red edge, and near-infrared wavelengths, as well as the nitrogen reflectance and greenness indices, which are sensitive to forest diseases. The study used random forest and spectral angle mapper approaches to classify disease severity, achieving an overall accuracy of 90.1% and a kappa coefficient of 0.87. The findings suggest that this method could be applicable to other forest types for timely and accurate disease detection.

[36] explored the use of micro-UAVs for monitoring apple quality and maturity in orchards. The UAV, equipped with an 18-channel visible/near-infrared multispectral sensor, collected internal quality

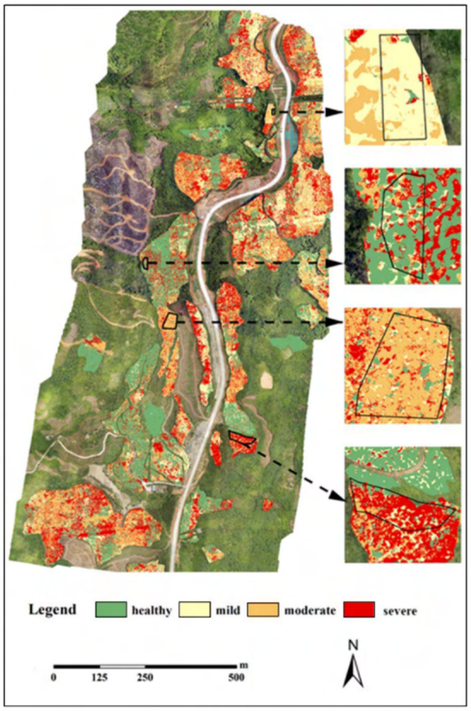


Figure 11: Spatial distributions of eucalyptus health conditions using random forest classifier based on the combination of three spectral bands and the nitrogen reflectance index. Tree health condition was defined according to the percentage of unhealthy leaves: 0 is healthy; 0%–30% is mild; 30%–60% is moderate; > 60% is severe [20].

information of apples at close range. A retractable push pole is used for spectral acquisition of apples on trees, as shwon in Fig.12. The spectral acquisition module, which includes six halogen lamps and a multispectral sensor, was placed inside a black shell to minimize the impact of sunlight variation. The module was pushed toward the apple to collect spectral data, using diffuse reflection. Data were processed using Multiple Scatter Correction (MSC) and Standard Normal Variate (SNV) pretreatment, combined with Partial Least Squares Regression (PLSR) to predict apple quality. The Streif index was used to assess apple maturity. Results showed that UAV detection did not affect apple growth and successfully collected internal quality data. The prediction model for apple maturity achieved a correlation coefficient of 0.8593 and a root mean square error of 0.0102, indicating the potential of UAVs for assessing apple quality and maturity in orchards [36]. Nevertheless, the designed spectral acquisition module can only adapt to a specific range of the apple size. For example, the apple is too small, which will lead to light leakage during acquisition [36]. Moreover, strong wind and rain will affect the operation of UAV [36].

[31] explored the use of visible and near-infrared spectral ranges to assess the vigor of sweet corn seeds under different processing conditions, including heat damage and artificial aging. Shown in Fig.13, the study employed two spectrometers with different bandwidths and variable selection algorithms to optimize the models for seed classification. Discrimination models were developed using full-wavelength data, with effective variables selected via competitive adaptive reweighted sampling.

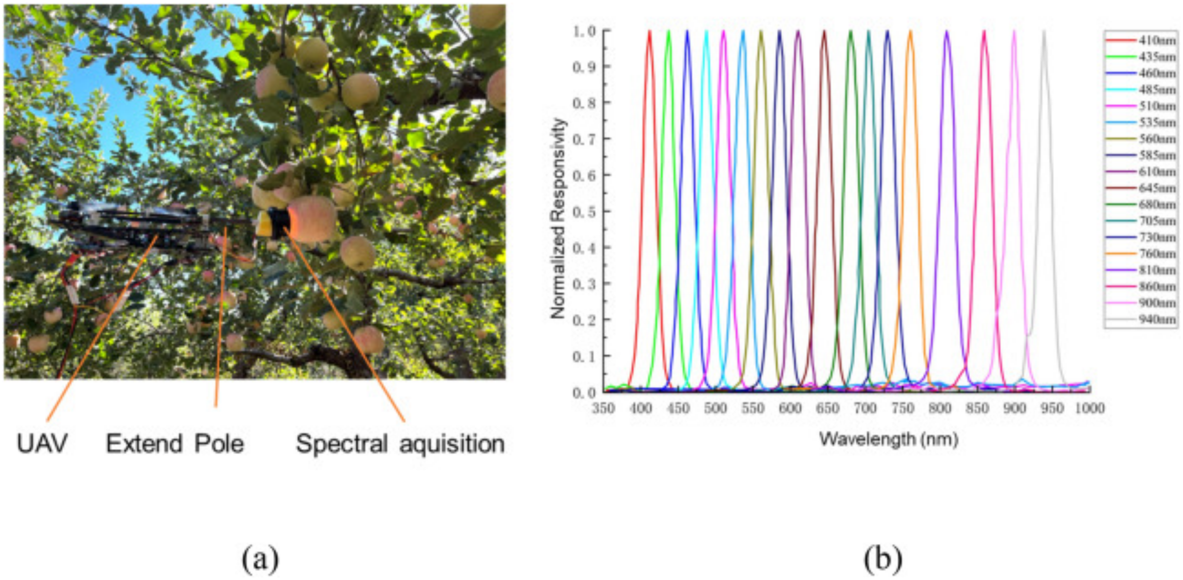


Figure 12: (a) UAVs with sensor; (b) Multispectral response range. [36]

The models based on both spectrometers achieved over 95% accuracy. The findings suggest that spectral ranges of 500–1100 nm or 1000–1850 nm can effectively discriminate seed vigor. Additionally, using effective variables extracted from the spectrometer reduced data processing requirements without compromising model accuracy, making the method more cost-efficient.

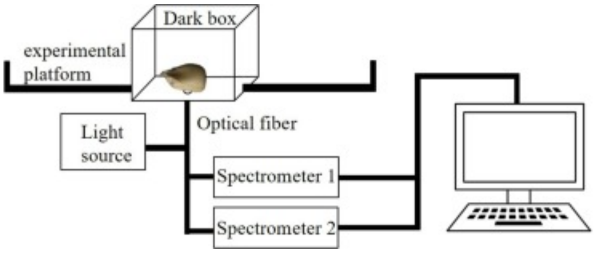


Figure 13: Schematic diagram of seed scanning structure

Furthermore, measuring soil water content (SWC) is an important and active area of research in precision agriculture, as water is essential for agriculture, particularly in water-scarce regions that have limited water availability [14]. [13] introduces a technique called Strobe for estimating soil moisture and electrical conductivity (EC) using Wi-Fi signals, by measuring the relative time of flight of Wi-Fi signals between multiple antennas, and the ratios of the amplitudes of the signals. Although experiments have shown that Strobe can accurately estimate soil moisture and EC at various moisture and salinity levels, the setup requires burying battery-powered antenna arrays under the soil, as shown in Fig.14. This process is cumbersome and labor-intensive, and the reliance on battery power poses limitations on long-term monitoring, as frequent replacements or recharging are needed to maintain functionality.

On the other hand, [5] evaluated the efficacy of thermal (TH) and multispectral (MS) cameras using

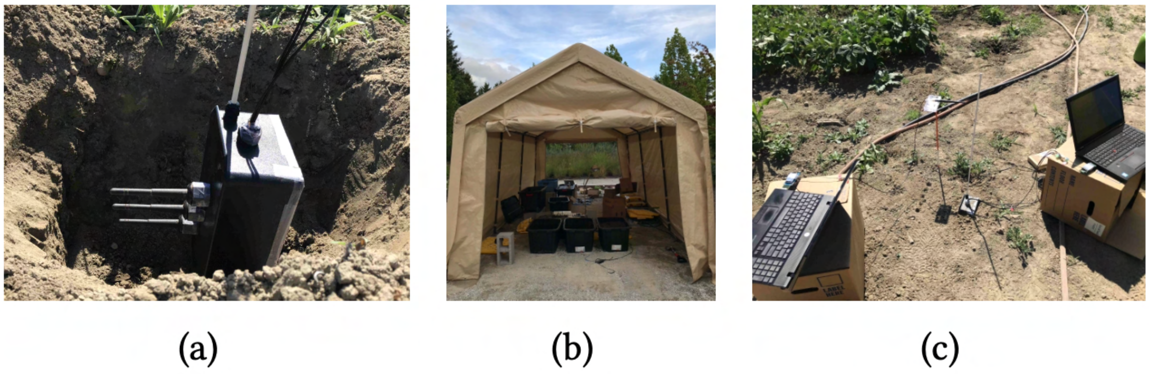


Figure 14: Soil measurement setup for multi-antenna system. (a) Receive antenna array on a waterproof box. (b) Tent with soil boxes. (c) Measurement setup on a farm. [13]

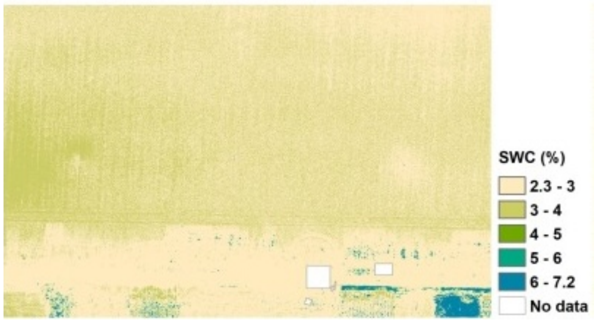


Figure 15: Estimation of soil water content (SWC) with multispectral [5].

a UAV-based system in SWC mapping using machine learning. The result shown that multispectral imagery effectively maps SWC, shown in Fig.15, with results demonstrating high model performance (R^2 up to 0.9). Through merged survey data, the study improved model accuracy, revealing that repeated surveys capture more soil moisture variability. Compared to [13] which requires burying the antenna arrays, [5] is more graceful, requires less setup and maintenance, and is an overall more established approach for estimating SWC.

2.4 Data Collection Methods and Types in Agricultural Robotics

As mentioned in 2.1, the growing global population has increased food demand, challenging agriculture to maximize productivity with limited resources. Precision Agriculture (PA) addresses this by using IoT-based sensors to monitor crop conditions at various growth stages [27]. PA gathers extensive data on crop health, including water levels, temperature, soil nutrients, pest presence, chlorophyll content, and weather conditions, to guide agronomic decisions. For example, chlorophyll levels and soil characteristics help determine optimal fertilizer needs, ensuring precise nutrient application [27].

Agricultural Robots have shown to optimize farming inputs and reveal denied or concealed information [25]. [1] built a system consists of two main components: a greenhouse control system and an autonomous robotic system, "iPlant", shown in Fig.16. The greenhouse control system maintains optimal conditions using sensors and actuators to regulate temperature, humidity, lighting, and smoke detection, with daily SMS updates sent to the owner. Meanwhile, iPlant operates autonomously to plant seeds and water soil as needed, switching between these tasks with a line-following approach and a push-button control. Together, these systems ensure efficient plant care by automating climate control, planting, and irrigation, reducing the need for manual intervention.

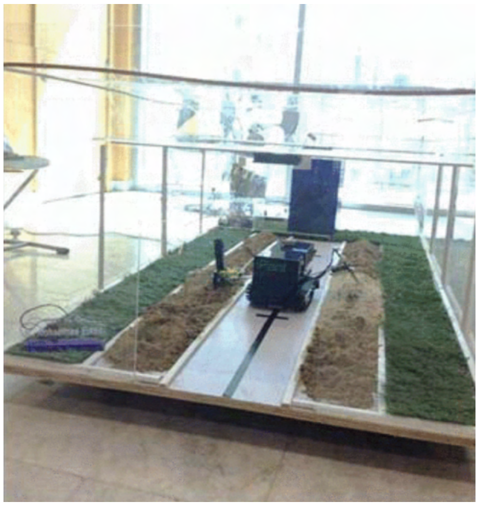


Figure 16: The greenhouse controlling system [1]

Similarly, [7] argues that collaborative robots can significantly reduce labor costs in organic agriculture, even on small to medium production scales. The SPECULARIA project utilizes microsensing solutions to create multimodal functional models of plants for autonomous monitoring, planning, and execution of agricultural tasks, as shown in Fig.17.

The project also explores replacing expensive specialized robots such as a Mobile Manipulator with a team of simpler robots, achieving the same efficiency through innovative planning solutions. The proposed SPECULARIA robotic system is shown in Fig.18. The robot uses a sensory setup to capture multispectral data in an indoor farmhouse workspace, shown in Fig.19. The setup includes an Intel RealSense RGB-D camera and a RedEdge-MX dual-camera system mounted on a custom design, moisture sensors are attached to end effectors to measure water conditions near the roots. This data is processed by a perception system to create a structural plant model that supports robotic manipulation planning. The model includes essential information for tasks such as harvesting, where fruit detection and leaf segmentation guide safe grasping, and for manual pollination, flower detection is used. Watering is planned based on vegetation indices, with safe execution ensured through a compliant control algorithm. The 3D functional plant model incorporates multimodal semantic data, enabling effective planning, execution of treatments, and tracking of plant development over time [7]. Lastly, experiments from the study show that using a stationary manipulator with supporting unmanned ground vehicles (UGVs) offers several advantages, both economically and in terms of long-term deployment. Larger UGVs that carry robotic arms consume more energy, requiring frequent battery recharges and resulting in idle time. Additionally, these larger UGVs are more expensive than smaller ones, making the system with multiple smaller UGVs more cost-effective [7].

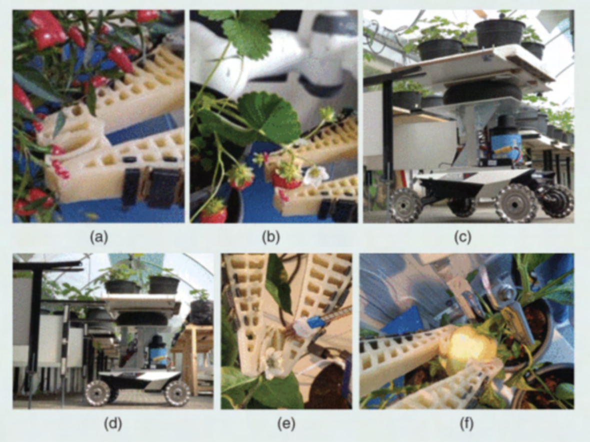


Figure 17: The dual-arm manipulation system picking (a) and (f) peppers, (e) picking strawberries and pollinating flowers. Transporting plants between growth pods (c) and (d) [7].

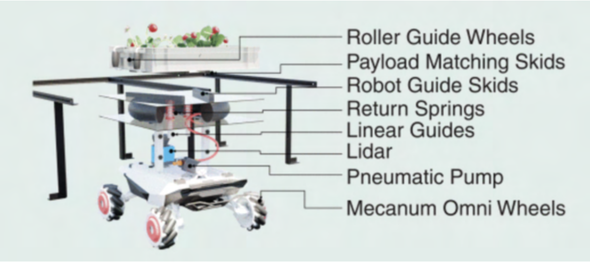


Figure 18: The robot used in SPECULARIA project

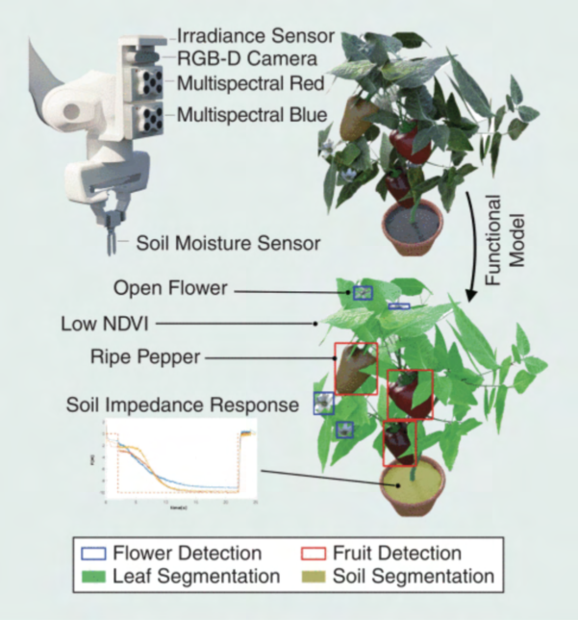


Figure 19: A multimodal sensory setup in an eye-in-hand configuration

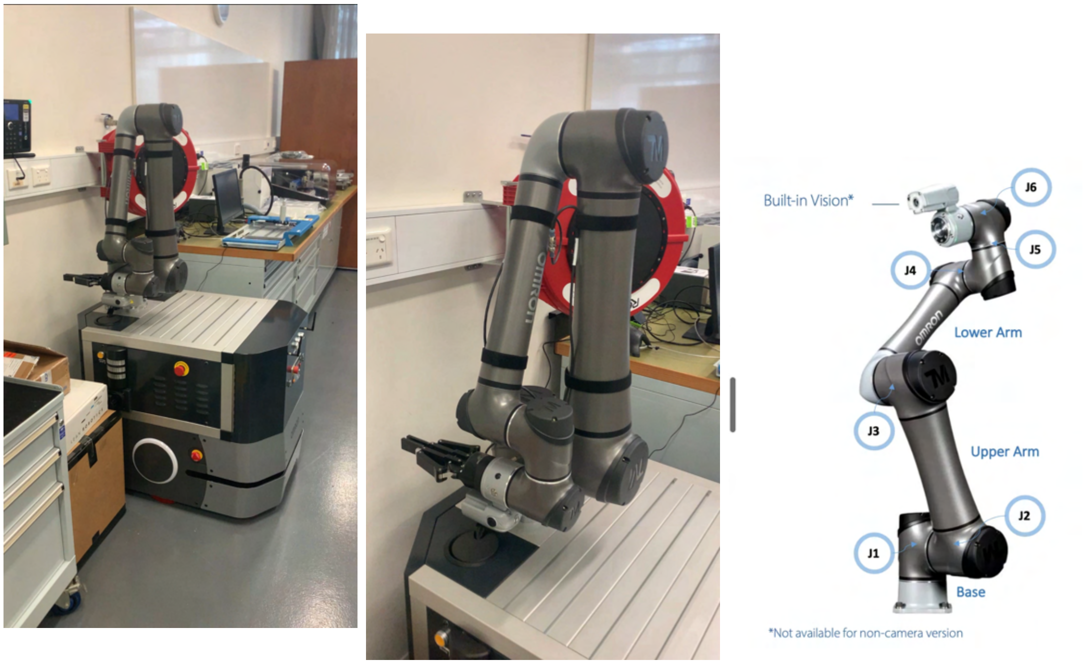
3 Methodology

3.1 Interfacing with MoMa

This project utilises the LD250 Mobile Manipulator (MoMa) from Omron, shown in Fig. 20. The LD250 MoMa comprises two main components: the LD250 Autonomous Mobile Robot (AMR) and the TM12S robotic arm.

Omron provides two main software interfaces for interacting with the MoMa. *TMFlow* is a visual programming environment that simplifies robotic arm control through user-friendly flowcharts and pre-configured modules. For AMR navigation and task automation, *MobilePlanner* is used.

Beneath these software layers lie lower-level command-line interfaces: the Advanced Robotics Control Language (ARCL) for the AMR, and TM Expression for the robotic arm. These interfaces enable direct communication with the respective hardware drivers.



(a) LD250 MoMa in a laboratory setting

(b) TM12S 6-DOF robotic arm in a laboratory setting

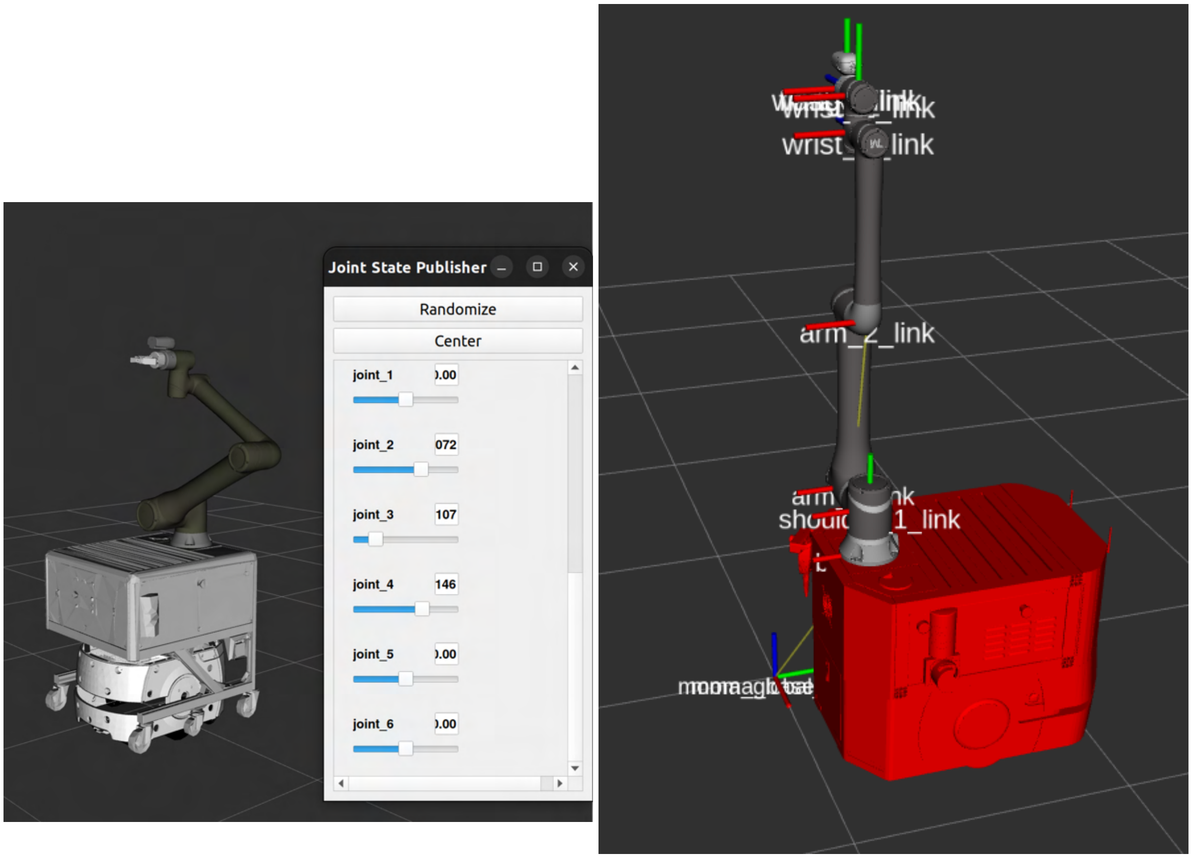
(c) TM12S Robotic Arm (image courtesy of Omron)

Figure 20: Omron LD250 MoMa system setup

To enable integration with ROS2, it is necessary to bypass the proprietary software interfaces and instead communicate directly with the hardware drivers. This integration requires addressing several technical challenges, outlined in the following sections.

3.1.1 MoMa Visualisation and Digital Twin

Open-source packages developed by Omron APAC[4][5][6] were used to visualise the MoMa in *RViz* and create a digital twin. However, the provided AMR and controller box models were outdated. To reflect the current hardware setup, custom STL models were imported, as shown in Fig. 21b. Although this improved the visual representation, a fully accurate digital twin was not achieved.



joint_1).00
joint_2 072
joint_3 107
joint_4 146
joint_5).00
joint_6).00

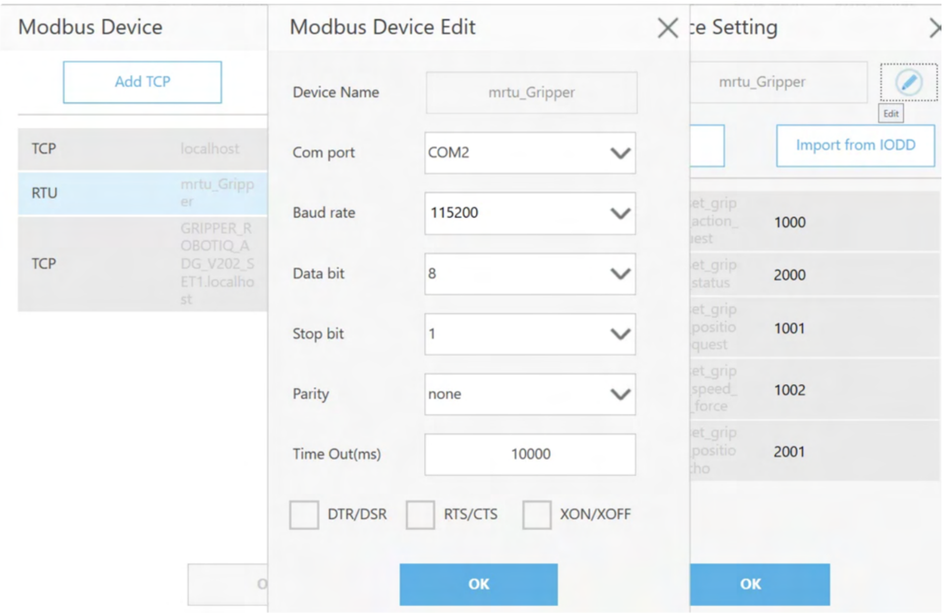
(a) Digital twin with outdated AMR and controller box (b) Updated visualisation with correct STL models

Figure 21: MoMa visualisation in *RViz*

[4] https://github.com/OmronAPAC/Omron_AMR_ROS2/tree/master
[5] https://github.com/OmronAPAC/Omron_TM_ROS2
[6] https://github.com/OmronAPAC/Omron_MoMa_ROS2

3.1.2 Gripper Control via Modbus

The Robotiq AGC-OMRON-KIT-140 adaptive gripper communicates using the Modbus RTU protocol over an RS485 serial connection to the PLC. Normally, this is abstracted through *TMFlow*, where preconfigured modules internally call the `modbus_write()` function. This function relies on a device entry defined in TMFlow's "Modbus Device Setting" (see Fig. 22).



TCP	localhost
RTU	mrtu_Gripper
TCP	GRIPPER_ROBOTIQ_ADG_V202_SET1.localhost

Device Name: mrtu_Gripper
Com port: COM2
Baud rate: 115200
Data bit: 8
Stop bit: 1
Parity: none
Time Out(ms): 10000
□ DTR/DSR □ RTS/CTS □ XON/XOFF

et_grip action_ est	1000
et_grip status	2000
et_grip positio quest	1001
et_grip speed_ force	1002
et_grip positio ho	2001

Figure 22: Modbus device configuration in TMFlow

Although TMFlow is not entirely bypassed, the gripper can be controlled directly via command line by sending TM Expression scripts through ROS2 services. The following example writes data `0x090000FF` to the Register Output at address `0x03E8` for the Modbus slave `0x0009`, referenced by the alias `"mrtu_Gripper"`.

```
ros2 service call /send_script tm_msgs/srv/SendScript "{id: '1', script:
 'modbus_write(\"mrtu_Gripper\", 0x0009, \"RO\", 0x03E8, 0x090000FF)'}"
```

Further documentation and scripts are available in Appendix A.

3.1.3 Controlling the AMR via ARCL

The `Omron_AMR_ROS2`[7] package provides a ROS2 service-client interface for sending ARCL commands to the AMR. For example, the following service call queues a task that moves the AMR 100 mm forward:

```
ros2 service call /arcl_api_service om_aiv_msg/srv/ArclApi "{command: 'dotask move 100',
 line_identifier: 1}"
```

[7] `https://github.com/OmronAPAC/Omron_AMR_ROS2/tree/master`

3.1.4 Bypassing TMFlow to Control the Robotic Arm

By default, the TM12S arm executes a TMFlow program that simplifies robot control via flowcharts. However, this abstraction prevents direct access to the arm and, critically, places the AMR in Emergency Stop (EStop) mode when the arm is active.

To regain full control, the default program was modified to include a strategically placed *Listen* node (Fig. 23). This node opens a TCP socket and listens for incoming TM Expression commands. Additionally, it prevents EStop from being triggered by disabling certain state transitions, effectively decoupling the robotic arm from the AMR.

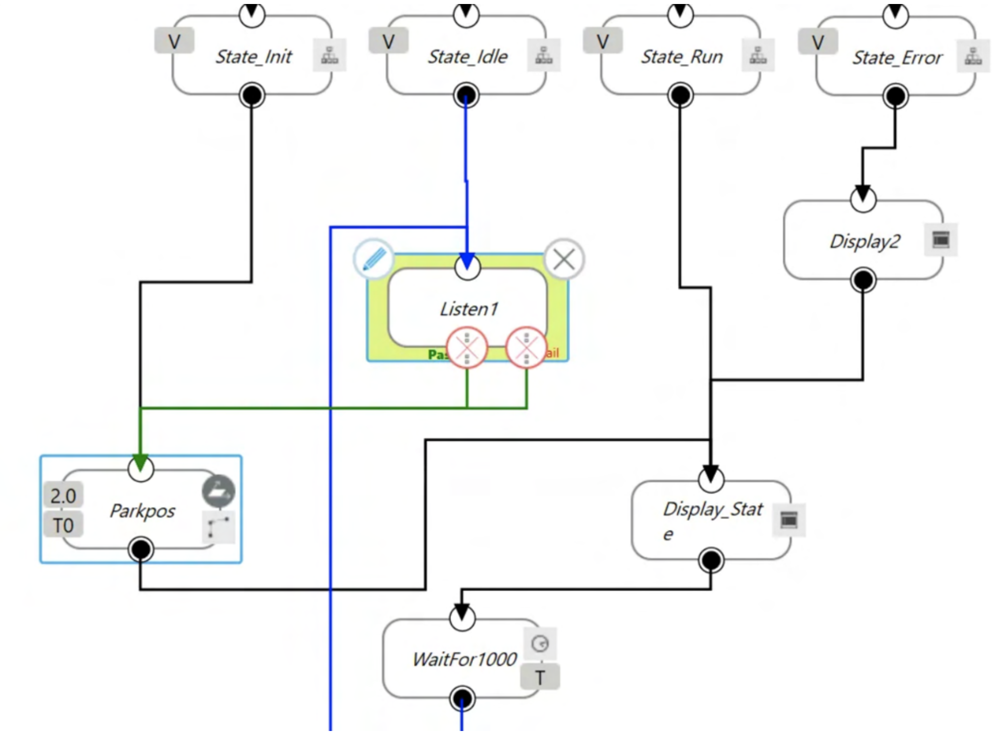
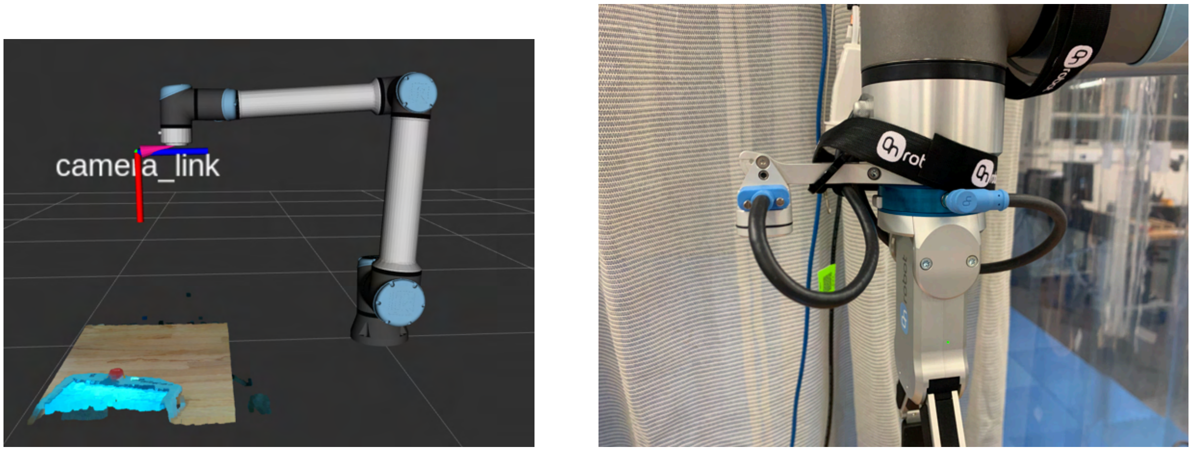


Figure 23: Modified TMFlow with TCP listener for direct arm control

3.2 Hand-Eye Calibration

Hand-eye calibration establishes a transformation between the camera frame and the robot's base link, allowing 3D image coordinates (with depth) to be mapped into the robot's workspace for manipulation tasks. This transformation is illustrated in Fig. 24a.

The calibration setup uses a UR10e 6-DOF robotic arm from Universal Robots and a RealSense D415 depth camera mounted on the end effector, shown in Fig. 24b.



(a) Visualisation of the camera_link frame

(b) UR10e with RealSense D415 camera mounted

Figure 24: Hand-eye calibration setup

The `moveit_calibration`[8] package is used to perform the calibration via a GUI plugin that simplifies the process. Further details are available in Appendix B.

3.3 Motion Planning and Planner Selection

The MoveIt! motion planning framework was selected for its deep integration with ROS2 and user-friendly configuration tools. It supports various planning backends, and the Open Motion Planning Library (OMPL) was used for this project due to its wide selection of planning algorithms.

OMPL provides:

- Single-query planners (e.g., Rapidly-exploring Random Trees (RRT))
- Multi-query planners (e.g., Probabilistic Roadmap Method (PRM))
- Optimising planners (e.g., RRT#)

It is important to note that not all OMPL planners are supported in MoveIt!, so selection was constrained to those available through its interface.

[8] `https://github.com/moveit/moveit_calibration`

3.4 Object Detection using YOLOv11

You Only Look Once (YOLO) is a popular real-time object detection framework. For this project, `YOLOv11` was selected to enable vision-based pick-and-place functionality.

Initial testing revealed that the default model performed poorly in detecting apples under laboratory lighting conditions. To address this, a custom model was trained using data from an online dataset[9] combined with 50 images captured by the lab's depth camera.

The resulting model, based on the `YOLOv11m` architecture, performed well in controlled environments (see Fig. 26), though it lacked robustness in other settings due to the dataset's strong bias toward lab images. Training results are summarised in Fig. 25.

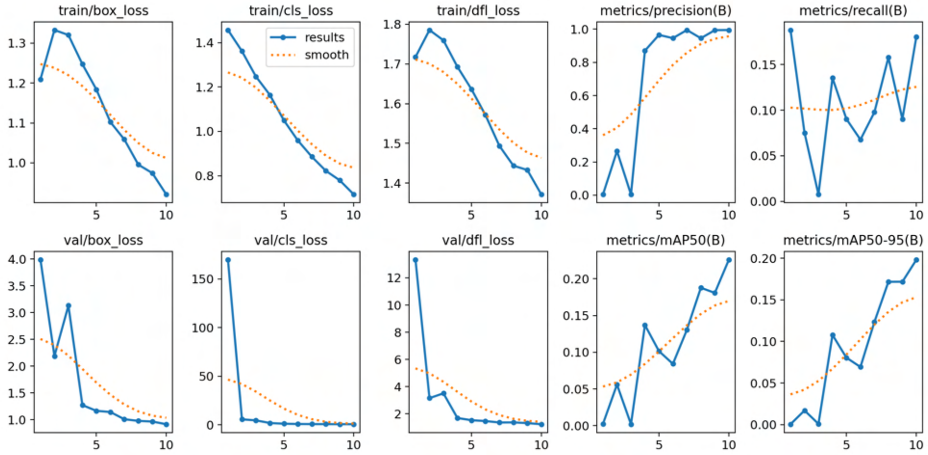


Figure 25: Custom YOLOv11 training results

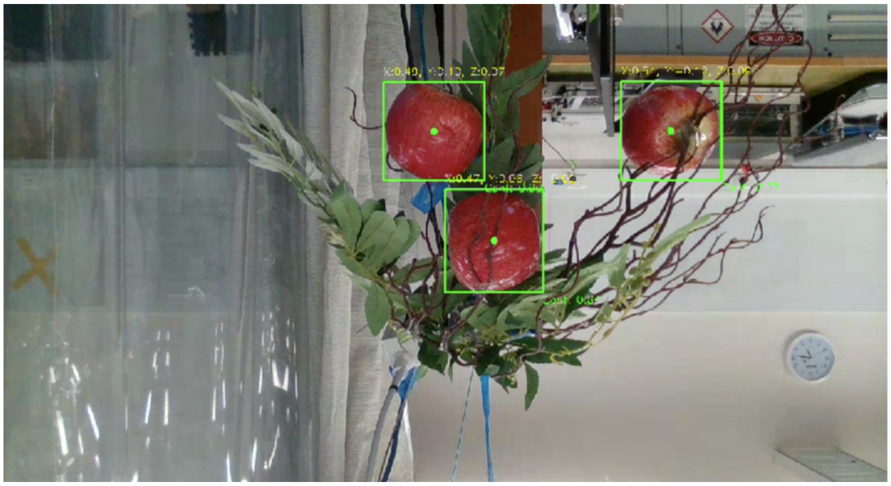


Figure 26: Example of successful detection in lab environment

[9] `https://universe.roboflow.com/the-pennsylvania-state-university/apple-detection-yolo`

4 Results and Discussion

4.1 Results

4.1.1 Interfacing with MoMa through ROS2[10]

As described in the Methodology section, the team successfully bypassed Omron's proprietary user interfaces, namely *MobilePlanner* and *TMFlow* in order to control MoMa via a custom ROS2 interface. This interface allows ARCL and TM Expression commands to be sent directly to the AMR and TM12S robotic arm, respectively. A demonstration program was developed to showcase the following:

- Independent control of the AMR and the TM12S robotic arm
- Independent control of the gripper
- Automation of sequences through scripted actions
- Bidirectional communication of ARCL commands with the PLC

A demonstration video can be accessed via the following link:
`https://www.youtube.com/shorts/6QjGDZB__Jw`. The source code for the demonstration script is provided in Appendix C.

4.1.2 Vision-Based Pick and Place on a Horizontal Surface Across Various Heights[11]

Figure 27 illustrates the experimental setup for vision-guided pick-and-place tasks on a horizontal surface at varying heights. The RRTConnect planning algorithm from OMPL was used for trajectory planning. The system achieved a consistent cycle time of 17–20 seconds for the entire process:
`Initial Detection → Subsequent Detection and Localisation → Pick & Place`.
A demonstration video is available at `https://youtu.be/pAT8YOUHFJc`.

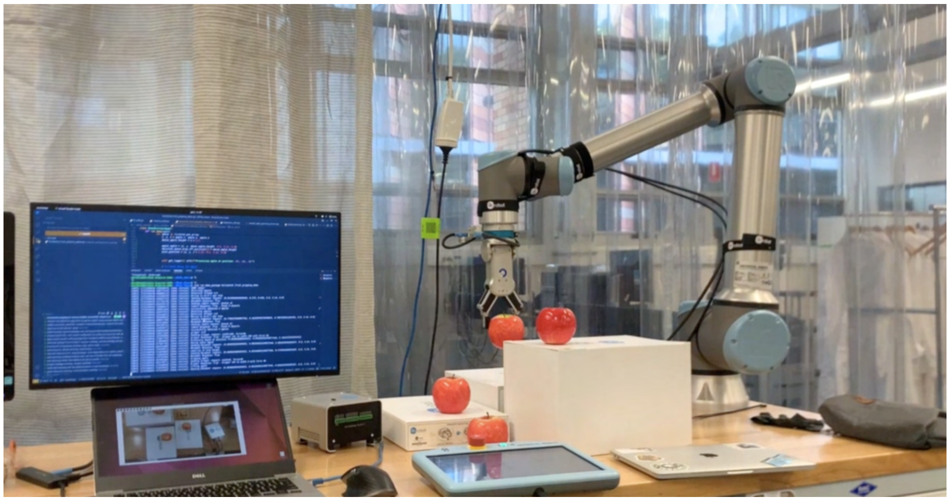


Figure 27: Horizontal pick-and-place demonstration setup

[10] Source code available at `https://github.com/DaviddNie/omron_moma_ros2`
[11] Source code available at `https://github.com/DaviddNie/UR10e_vision_based_fruit_harvesting`

4.1.3 Vision-Based Fruit Harvesting[12]

Figure 28 shows the setup for simulating vision-based fruit harvesting in a realistic vertical orientation. After extensive testing of OMPL planning algorithms, the Transition-based RRT (TRRT) planner was selected due to its effectiveness in finding short, low-cost paths. However, the overall cycle time, which includes scanning, detecting, localising, picking, and placing averaged approximately 75 seconds, mainly due to suboptimal motion planning performance.

An attempt to improve planning with the RRT# optimisation planner yielded significantly better paths. However, it required around four minutes of computation time, increasing the total cycle to approximately five minutes.

The demonstration video is available at `https://youtu.be/1r7PfkH8pU8`.

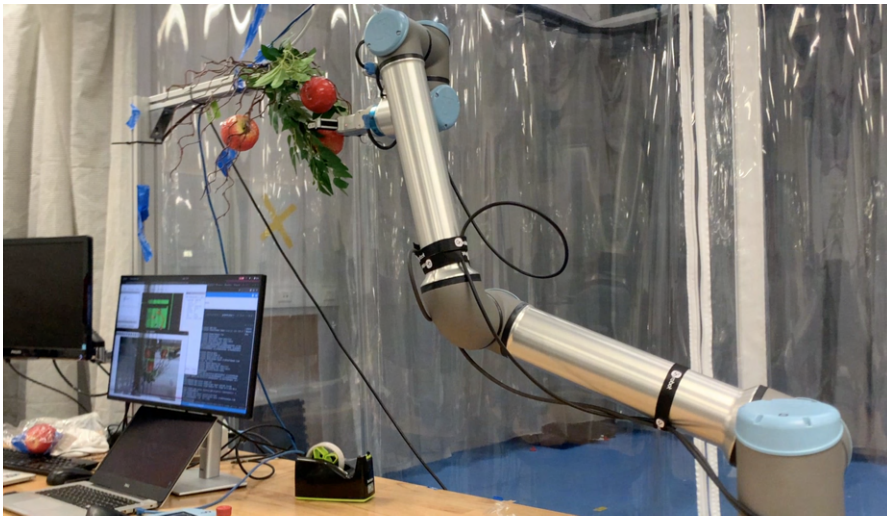


Figure 28: Vertical pick-and-place demonstration setup

4.1.4 Evaluation of Project Hypotheses

Two hypotheses were proposed at the start of the project:

1. A Human-Robot System (HRS), where the operator defines high-level navigation parameters, provides advantages over fully autonomous systems in terms of design simplicity, operational robustness, and adaptability to unstructured environments, thereby enabling more reliable and efficient plant data collection.

2. In vision-based fruit harvesting, manually defining the robot's position and orientation relative to the plant improves system efficiency over fully autonomous approaches, due to better spatial optimisation for task execution.

Due to unexpected hardware failure during the latter stages of the project, the team continued experimentation using a UR10e 6-DOF robotic arm from Universal Robots.

[12] Source code available at `https://github.com/DaviddNie/UR10e_vision_based_fruit_harvesting`

Autonomous fruit harvesting systems must handle varying lighting conditions and dynamically identify optimal robot placement and gripper orientation, this is particularly challenging when fruits are occluded by foliage. The experiments demonstrated that optimal joint configurations (as seen in horizontal pick-and-place tasks) significantly enhance planning efficiency compared to vertical scenarios, where the robot is more prone to singularities. These singularities often force planners to compute longer paths to avoid unsafe configurations.

The results align with findings from [24], indicating that fully autonomous systems in agriculture are neither cost-effective nor productive enough to replace semi-autonomous solutions or human labour. Regarding HRS, allowing operators to define parameters such as planner type, entry orientation, and gripper configuration offers significant performance improvements in specific scenarios. However, compared to specialised agricultural machinery, the benefits of HRS may diminish, especially when considering cost and operational efficiency at scale.

4.2 Future Work and Recommendations

4.2.1 MoMa

The ROS2 interface developed for MoMa lays the groundwork for future research directions:

1. **SLAM Integration:** Access to the LiDAR sensor is currently restricted, as it is integrated within MobilePlanner and not exposed through ARCL. For autonomous operation and localisation, accessing and integrating this data is essential.

2. **Fruit Harvesting with AMR Mobility:** The current results suggest that certain configurations may lead to singularities. Future work could explore repositioning the AMR during tasks to help maintain optimal joint configurations for the robotic arm.

4.2.2 Motion Planning and Cycle Time

While the planner performed well in horizontal pick-and-place tasks, vertical operations remain suboptimal. With an improved planner, the cycle time could potentially be reduced to 30 seconds, half of the current duration. Unexplored planning options include:

1. Alternative planning frameworks such as CHOMP, STOMP, and the Pilz industrial motion planner

2. Integration of more advanced OMPL planners, such as RRT, CForest, and others

4.2.3 Hardware Considerations

One major failure point was poor gripper entry angles, often resulting in objects being pushed away rather than grasped. This can be addressed by using a more task-specific end-effector, rather than attempting to compensate through planner behaviour alone.

Historically, specialised machinery has been developed for agricultural tasks. Instead of optimising the fruit harvesting cycle time using robotic arms, it may be more beneficial to rethink the harvesting process entirely, i.e. developing more efficient and novel harvesting mechanisms tailored to automation.

5 Conclusion

The proof-of-concept system demonstrates that vision-based fruit harvesting is technically feasible but not yet operationally efficient. Our experiments revealed a fourfold performance difference between task orientations, with vertical harvesting requiring 75 seconds per cycle compared to 17-20 seconds for horizontal tasks. These results validate both our hypothesis about human-robot system advantages and [24]'s critique of autonomous solutions. The efficiency gap primarily stems from motion planning limitations and gripper design challenges, similar to those identified in [19] for mmWave applications requiring ideal conditions.

Three key findings emerge from this work. First, the dramatic variation in motion planning performance based on orientation confirms the environment-dependent nature of agricultural robotics noted by [3]. Second, the system's need for manual pose initialization supports [29]'s 80/20 rule regarding optimal task partitioning between humans and machines. Third, frequent gripper misalignments corroborate [7]'s findings about the superiority of specialized end-effectors.

Future efficiency improvements could focus on three areas: SLAM integration to overcome localization bottlenecks identified in [26], development of fruit-specific grippers to significantly improve grasping reliability, and implementation of optimization planners like RRT*. However, as [25] notes, such incremental enhancements may prove insufficient to match dedicated harvesting machinery. A more fundamental redesign of robotic harvesting systems, potentially incorporating collaborative human-robot workflows, may be necessary to achieve commercial viability.

References

[1] Bashayer Al-Beeshi, Bashayer Al-Mesbah, Sara Al-Dosari, and Mohammed El-Abd. iplant: The greenhouse robot. In *2015 IEEE 28th Canadian conference on electrical and computer engineering (CCECE)*, pages 1489–1494. IEEE, 2015.

[2] Javier Baluja, Maria P Diago, Pedro Balda, Roberto Zorer, Franco Meggio, Fermin Morales, and Javier Tardaguila. Assessment of vineyard water status variability by thermal and multispectral imagery using an unmanned aerial vehicle (uav). *Irrigation science*, 30:511–522, 2012.

[3] Avital Bechar and Clément Vigneault. Agricultural robots for field operations: Concepts and components. *Biosystems Engineering*, 149:94–111, 2016.

[4] László Bertalan, Imre Holb, Angelika Pataki, Gábor Négyesi, Gergely Szabó, Annamária Kupásné Szalóki, and Szilárd Szabó. Uav-based multispectral and thermal cameras to predict soil water content–a machine learning approach. *Computers and Electronics in Agriculture*, 200:107262, 2022.

[5] László Bertalan, Imre Holb, Angelika Pataki, Gábor Négyesi, Gergely Szabó, Annamária Kupásné Szalóki, and Szilárd Szabó. Uav-based multispectral and thermal cameras to predict soil water content – a machine learning approach. *Computers and Electronics in Agriculture*, 200:107262, 2022.

[6] Robert Bogue. Sensors key to advances in precision agriculture. *Sensor Review*, 37(1):1–6, 2017.

[7] Marsela Car, Barbara Arbanas Ferreira, Jelena Vuletic, and Matko Orsag. Structured ecological cultivation with autonomous robots in agriculture: Toward a fully autonomous robotic indoor farming system. *IEEE Robotics Automation Magazine*, 30(4):77–87, 2023.

[8] Mark Cardamis, Hong Jia, Hao Qian, Wenyao Chen, Yihe Yan, Oula Ghannoum, Aaron Quigley, Chung Tung Chou, and Wen Hu. Leafeon: Towards accurate, robust and low-cost leaf water content sensing using mmwave radar. *arXiv preprint arXiv:2410.03680*, 2024.

[9] R Ceres, Jose L Pons, AR Jimenez, JM Martin, and L Calderon. Design and implementation of an aided fruit-harvesting robot (agribot). *Industrial Robot: An International Journal*, 25(5):337–346, 1998.

[10] Keun Ha Choi, Sang Kwon Han, Sang Hoon Han, Kwang-Ho Park, Kyung-Soo Kim, and Soohyun Kim. Morphology-based guidance line extraction for an autonomous weeding robot in paddy fields. *Computers and Electronics in Agriculture*, 113:266–274, 2015.

[11] Keun Ha Choi, Sang Kwon Han, Sang Hoon Han, Kwang-Ho Park, Kyung-Soo Kim, and Soohyun Kim. Morphology-based guidance line extraction for an autonomous weeding robot in paddy fields. *Computers and Electronics in Agriculture*, 113:266–274, 2015.

[12] Itamar Dar, Yael Edan, and Avital Bechar. An adaptive path classification algorithm for a pepper greenhouse sprayer. In *2011 Louisville, Kentucky, August 7-10, 2011*, page 1. American Society of Agricultural and Biological Engineers, 2011.

[13] Jian Ding and Ranveer Chandra. Towards low cost soil sensing using wi-fi. In *The 25th Annual International Conference on Mobile Computing and Networking*, MobiCom '19, New York, NY, USA, 2019. Association for Computing Machinery.

[14] Paolo D'Odorico, Davide Danilo Chiarelli, Lorenzo Rosa, Alfredo Bini, David Zilberman, and Maria Cristina Rulli. The global value of water in agriculture. *Proceedings of the national academy of sciences*, 117(36):21985–21993, 2020.

[15] Yuval Garini, Ian T Young, and George McNamara. Spectral imaging: principles and applications. *Cytometry part a: the journal of the international society for analytical cytology*, 69(8):735–747, 2006.

[16] Ibrahim A Hemadeh, Katla Satyanarayana, Mohammed El-Hajjar, and Lajos Hanzo. Millimeter-wave communications: Physical channel models, design considerations, antenna constructions, and link-budget. *IEEE Communications Surveys & Tutorials*, 20(2):870–913, 2017.

[17] Natalia A Hoog, Tomas Evert van den Berg, and Harijot Singh Bindra. A 60 ghz pulsed coherent radar for online monitoring of the withering condition of leaves. *Sensors and Actuators A: Physical*, 343:113693, 2022.

[18] Mohamed Kerkech, Adel Hafiane, and Raphael Canals. Vddnet: Vine disease detection network based on multispectral images and depth map. *Remote Sensing*, 12(20):3305, 2020.

[19] Yumeng Liang, Anfu Zhou, Huanhuan Zhang, Xinzhe Wen, and Huadong Ma. Fg-liquid: A contact-less fine-grained liquid identifier by pushing the limits of millimeter-wave sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(3):1–27, 2021.

[20] Kuo Liao, Fan Yang, Haofei Dang, Yunzhong Wu, Kunfa Luo, and Guiying Li. Detection of eucalyptus leaf disease with uav multispectral imagery. *Forests*, 13(8):1322, 2022.

[21] Wentao Liu, Chenglin Wang, De Yan, Weilin Chen, and Lufeng Luo. Estimation of characteristic parameters of grape clusters based on point cloud data. *Frontiers in Plant Science*, 13:885167, 2022.

[22] David J Mulla. Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. *Biosystems engineering*, 114(4):358–371, 2013.

[23] Luiz F. P. Oliveira, António P. Moreira, and Manuel F. Silva. Advances in agriculture robotics: A state-of-the-art review and challenges ahead. *Robotics*, 10(2):52, 2021.

[24] Marsela Polic, Antun Ivanovic, Bruno Maric, Barbara Arbanas, Jelena Tabak, and Matko Orsag. Structured ecological cultivation with autonomous robots in indoor agriculture. In *2021 16th International Conference on Telecommunications (ConTEL)*, pages 189–195. IEEE, 2021.

[25] Redmond R Shamshiri, Cornelia Weltzien, Ibrahim A Hameed, Ian J Yule, Tony E Grift, Siva K Balasundram, Lenka Pitonakova, Desa Ahmad, and Girish Chowdhary. Research and development in agricultural robotics: A perspective of digital farming. 2018.

[26] Raphaël Rouveure, Patrice Faure, and Marie-Odile Monod. Pelican: Panoramic millimeter-wave radar for perception in mobile robotics applications, part 1: Principles of fmcw radar and of 2d image construction. *Robotics and Autonomous Systems*, 81:1–16, 2016.

[27] Uferah Shafi, Rafia Mumtaz, José García-Nieto, Syed Ali Hassan, Syed Ali Raza Zaidi, and Naveed Iqbal. Precision agriculture techniques and practices: From considerations to applications. *Sensors*, 19(17), 2019.

[28] Matthew Smithson, Melanie Po-Leen Ooi, Lauren Gris, Ye Chow Kuang, Merilyn Manley-Harris, and Shen Hin Lim. Investigating the use of low-cost and low-power millimeter wave radar to improve quality of tomato harvesting. In *2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pages 1–6. IEEE, 2021.

[29] Anthony Stentz, Cristian Dima, Carl Wellington, Herman Herman, and David Stager. A system for semi-autonomous tractor operations. *Autonomous Robots*, 13:87–104, 2002.

[30] Bram van Berlo, Amany Elkelany, Tanir Ozcelebi, and Nirvana Meratnia. Millimeter wave sensing: A review of application pipelines and building blocks. *IEEE Sensors Journal*, 21(9):10332–10368, 2021.

[31] Yali Wang, Yankun Peng, Xin Qiao, and Qibin Zhuang. Discriminant analysis and comparison of corn seed vigor based on multiband spectrum. *Computers and Electronics in Agriculture*, 190:106444, 2021.

[32] Chenshu Wu, Feng Zhang, Beibei Wang, and KJ Ray Liu. msense: Towards mobile material sensing with a single millimeter-wave radio. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(3):1–20, 2020.

[33] Dongbo Xie, Liang Chen, Lichao Liu, Liqing Chen, and Hai Wang. Actuators and sensors for application in agricultural robots: A review. *Machines*, 10(10):913, 2022.

[34] Yali Zhang, Dehua Zhao, Hanchao Liu, Xinrong Huang, Jizhong Deng, Ruichang Jia, Xiaoping He, Muhammad Naveed Tahir, and Yubin Lan. Research hotspots and frontiers in agricultural multispectral technology: Bibliometrics and scientometrics analysis of the web of science. *Frontiers in Plant Science*, 13:955340, 2022.

[35] Zhihong Zhang, Chaowei Huang, Xing Xu, Lizhe Ma, Zhou Yang, and Jieli Duan. Three-dimensional structure measurement for potted plant based on millimeter-wave radar. *Agriculture*, 13(11):2089, 2023.

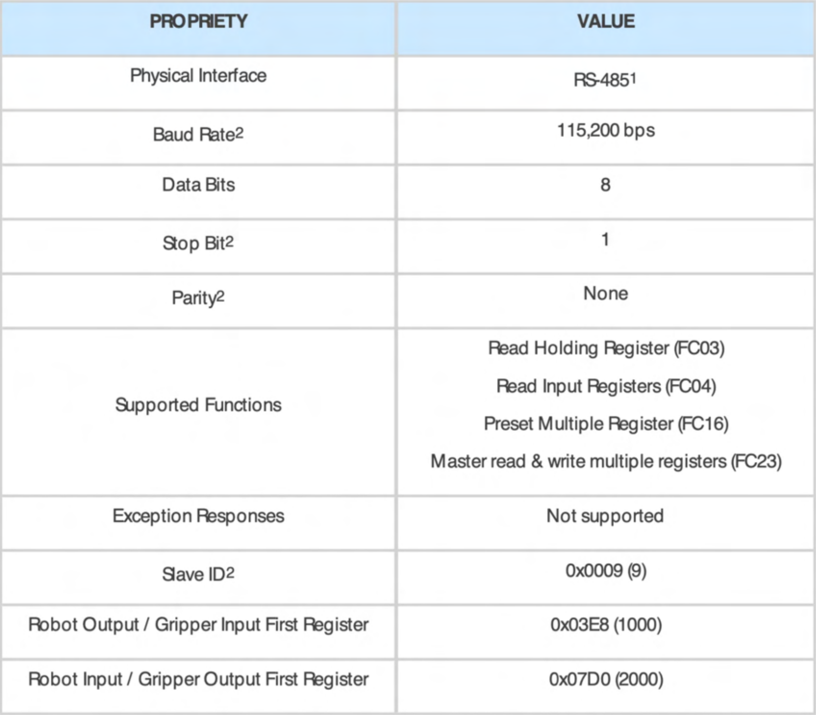
[36] Xinlong Zhao, Yankun Peng, Yongyu Li, Qinghui Guo, and Bingwei Wang. A flight sensing detector for apple maturity indexes in orchard. *Biosystems Engineering*, 230:470–479, 2023.

[37] Xiang Zhou, HB Zheng, XQ Xu, JY He, XK Ge, Xia Yao, Tao Cheng, Yan Zhu, WX Cao, and YC Tian. Predicting grain yield in rice using multi-temporal vegetation indices from uav-based multispectral and digital imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 130:246–255, 2017.

Appendices

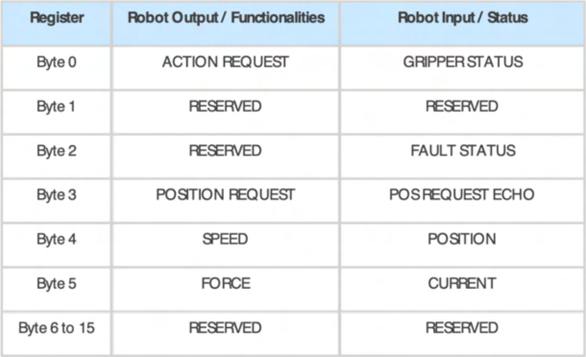
A Controlling the Gripper via Modbus

Figures 29 and 30 illustrate the properties and register configurations of the Robotiq AGC-OMRON-KIT-140 adaptive gripper. These screenshots are taken from the official Robotiq user manual[13].



PROPRIETY	VALUE
Physical Interface	RS-485[1]
Baud Rate[2]	115,200 bps
Data Bits	8
Stop Bit[2]	1
Parity[2]	None
Supported Functions	Read Holding Register (FC03) Read Input Registers (FC04) Preset Multiple Register (FC16) Master read & write multiple registers (FC23)
Exception Responses	Not supported
Slave ID[2]	0x0009 (9)
Robot Output / Gripper Input First Register	0x03E8 (1000)
Robot Input / Gripper Output First Register	0x07D0 (2000)

Figure 29: Gripper properties from the Robotiq manual



Register	Robot Output / Functionalities	Robot Input / Status
Byte 0	ACTION REQUEST	GRIPPER STATUS
Byte 1	RESERVED	RESERVED
Byte 2	RESERVED	FAULT STATUS
Byte 3	POSITION REQUEST	POS REQUEST ECHO
Byte 4	SPEED	POSITION
Byte 5	FORCE	CURRENT
Byte 6 to 15	RESERVED	RESERVED

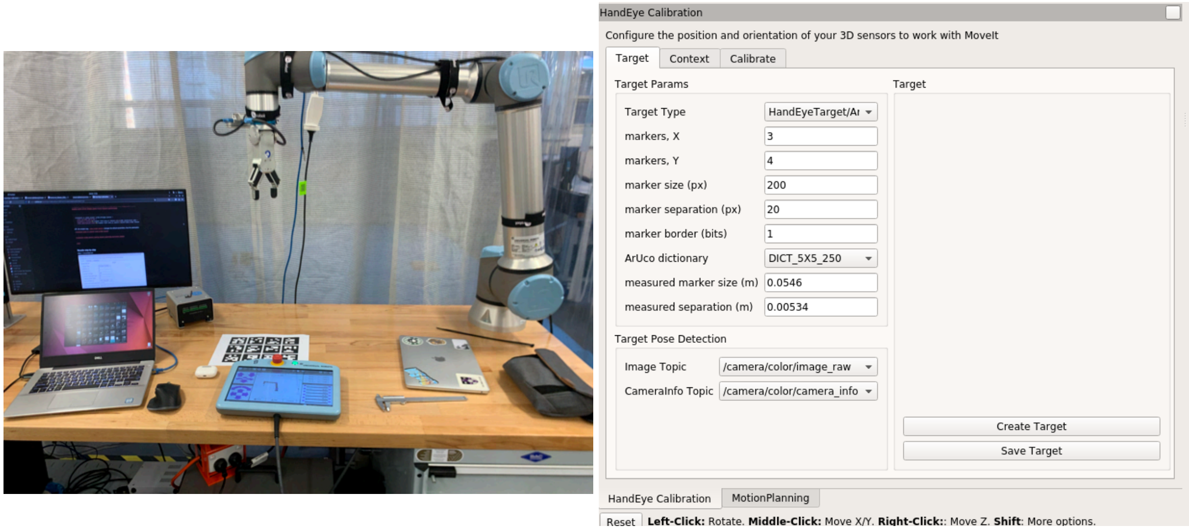
Figure 30: Gripper register settings from the Robotiq manual

[13] https://files.omron.eu/downloads/latest/manual/de/robotiq_2f-85_2f-140_for_omron_tm_seriesrobots_instruction_manual_de.pdf?v=1

B Hand-Eye Calibration

The `moveit_calibration` GUI plugin in *RViz* streamlines the hand-eye calibration process through a series of guided steps.

The first step involves generating and printing a marker array, such as an ArUco marker sheet, which is then placed in the workspace. This setup is shown in Fig. 31.



HandEye Calibration

Configure the position and orientation of your 3D sensors to work with MoveIt

Target Params

Target Type: HandEyeTarget/Ar
markers, X: 3
markers, Y: 4
marker size (px): 200
marker separation (px): 20
marker border (bits): 1
ArUco dictionary: DICT_5X5_250
measured marker size (m): 0.0546
measured separation (m): 0.00534

Target Pose Detection

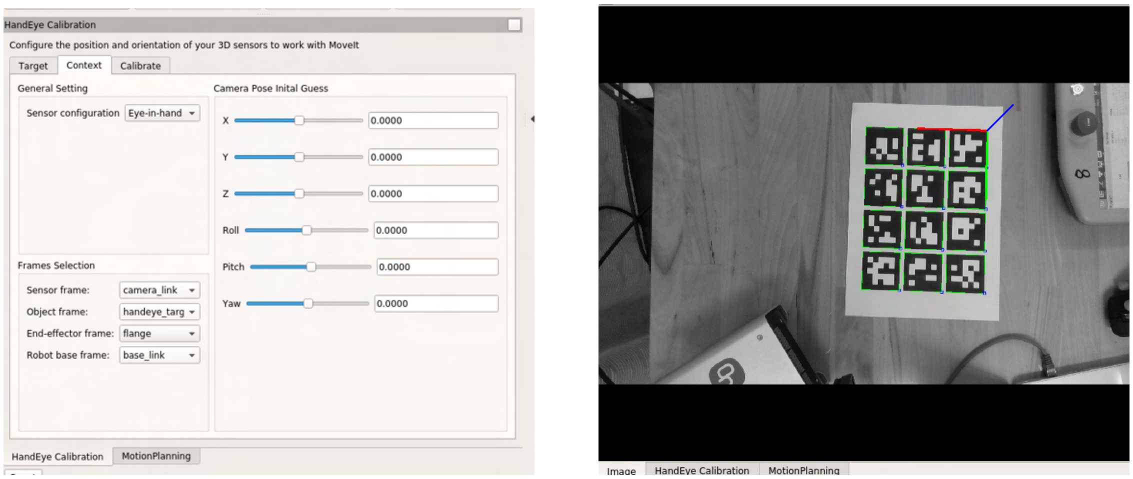
Image Topic: /camera/color/image_raw
CameraInfo Topic: /camera/color/camera_info

Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click:: Move Z. Shift: More options.

(a) Lab setup for hand-eye calibration (b) Graphical interface of the calibration plugin

Figure 31: Marker array setup for hand-eye calibration

Next, the user configures the sensor and robot frames in the calibration context, as shown in Fig. 32a. Once configured, the system begins detecting the ArUco markers and visualises them in *RViz* (Fig. 32b).



HandEye Calibration

Configure the position and orientation of your 3D sensors to work with MoveIt

General Setting

Sensor configuration: Eye-in-hand

Camera Pose Inital Guess

X: 0.0000
Y: 0.0000
Z: 0.0000
Roll: 0.0000
Pitch: 0.0000
Yaw: 0.0000

Frames Selection

Sensor frame: camera_link
Object frame: handeye_targ
End-effector frame: flange
Robot base frame: base_link

(a) Sensor and frame configuration in the context menu (b) Visualisation of ArUco marker detection in RViz

Figure 32: Calibration context setup and marker detection

For a successful $AX = XB$ calibration, at least five distinct robot poses are required. In this equation, A represents the transformation between two end-effector poses, B is the corresponding camera transformation between the same observations (estimated via the marker array), and X is the unknown rigid transformation from the camera frame to the robot's flange. The plugin documentation recommends collecting between 12 and 15 poses for improved accuracy.

In this project, 20 pose samples were collected. The final estimated transform was exported as a static transform publisher, shown in the code snippet below. This transform is also visualised in Fig. 33.

```
<launch>
  <!-- The rpy in the comment uses the extrinsic XYZ convention, \
  which is the same as is used in a URDF. See
       http://wiki.ros.org/geometry2/RotationMethods and
       https://en.wikipedia.org/wiki/Euler_angles for more info. -->
  <!-- xyz="0.0199931 -0.0028582 -0.109958" rpy="3.12825 -3.12574 -3.13495" -->
  <node pkg="tf2_ros" type="static_transform_publisher" name="camera_link_broadcaster"
      args="0.0199931 -0.0028582 -0.109958   -0.0066963 -0.00790545 0.00337418 0.999941
      flange camera_link" />
</launch>
```

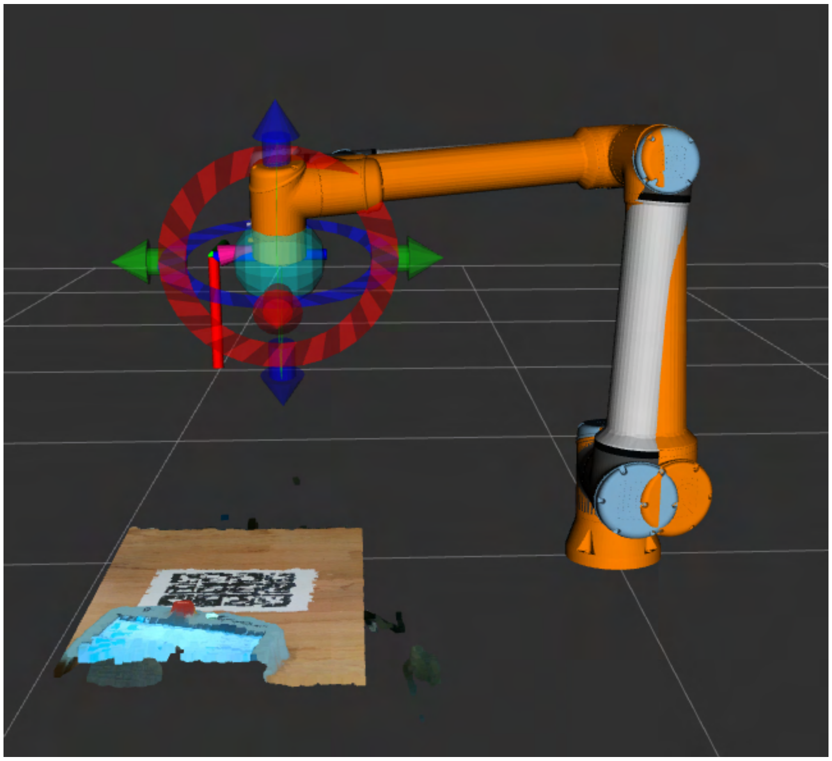


Figure 33: Hand-eye calibration visualised with point cloud from the depth camera

C Script Used in the MoMa Demonstration Video

The following script, together with three ROS2 service-client nodes, forms the core of the demonstration program[14]. This setup showcases the simplicity and efficiency of the workspace configuration.

```
import rclpy
from rclpy.node import Node
from om_aiv_msg.srv import AmrCmd
import time
import random
import string
from tm_msgs.srv import WirelessInfo


WAITAFTER_COMMAND = "waitAfter"
WAITCOMPLETE_COMMAND = "waitComplete"
QUERYJOB_COMMAND = "queryjob"
EXIT_COMMAND = "exit"
AMR = "AMR"
TM = "TM"
WAIT = "wait"

class DemoRoutine(Node):
 def __init__(self):
  super().__init__(’demo_routine’)
  self.client = self.create_client(AmrCmd, ’/amr_cmd’)
  self.wirelessClient = self.create_client(WirelessInfo, ’/wireless_info’)

  while not self.client.wait_for_service(timeout_sec=1.0):
   self.get_logger().info(’Amr_Arcl Service not available, waiting again...’)

  while not self.wirelessClient.wait_for_service(timeout_sec=1.0):
   self.get_logger().info(’wirelessClient Service not available, waiting again...’)

  self.get_logger().info(’DEMO ready!’)

 def process_request(self, type_, cmd, param1="", param2=""):
  if type_ == TM:
   response = self.send_blocking_request_tm(cmd, param1, param2)
   return response
```

[14] https://www.youtube.com/shorts/6QjGDZB__Jw

```
 if type_ == WAIT:
  time.sleep(float(cmd))
  return "Great Success~"

 return self.send_blocking_request_amr(cmd, param1, param2)

def send_blocking_request_amr(self, command, param1="", param2=""):
 # Create the request message
 request = AmrCmd.Request()
 request.command = command
 request.param1 = param1
 request.param2 = param2
 self.get_logger().info(f'Sending AMR request: {command} {param1} {param2}')

 future = self.client.call_async(request)

 while rclpy.ok():
  rclpy.spin_once(self, timeout_sec=0.1)

  if future.done():
   try:
    response = future.result()
    self.get_logger().info(f'Response: {response}')
    return response
   except Exception as e:
    self.get_logger().error(f'Service call failed: {e}')
    return None

def send_blocking_request_tm(self, command, param1="", param2=""):
 request = WirelessInfo.Request()
 request.command = command
 request.param1 = param1

 self.get_logger().info(f'Sending TM request: {command} {param1} {param2}')

 future = self.wirelessClient.call_async(request)

 while rclpy.ok():
  rclpy.spin_once(self, timeout_sec=0.1)

  if future.done():
   try:
    response = future.result()
    self.get_logger().info(f’Response: {response}’)
    return response
   except Exception as e:
    self.get_logger().error(f’Service call failed: {e}’)
    return None

def run_demo(self):

 token = ’’.join(random.choices(string.ascii_letters + string.digits, k=6))

 job_1_str = "job_1_" + token

 commands = [
  (AMR,"say", "starting"),
  (TM, "movearm", "home"),
  (AMR,"say", "moving forward"),
  (AMR,"move", "500"),
  (AMR,"say", "moving backward"),
  (AMR,"move", "-100"),
  (AMR,"say", "setheading"),
  (AMR,"setHeading", "180"),
  (AMR,"say", "move arm"),
  (TM, "movearm", "grip1"),
  (WAIT, "7"),
  (AMR,"say", "gripper engaging"),
  (TM, "grip"),
  (WAIT, "3"),
  (AMR,"say", "move arm"),
  (TM, "movearm", "release1"),
  (AMR,"say", "moving forward"),
  (AMR,"move", "300"),
  (WAIT, "7"),
  (AMR,"say", "release in 3 seconds"),
  (WAIT, "0.4"),
  (AMR,"say", "3"),
  (WAIT, "0.4"),
  (AMR,"say", "2"),
  (WAIT, "0.4"),
  (AMR,"say", "1"),
   (WAIT, "0.4"),
   (AMR,"say", "releasing"),
   (WAIT, "0.2"),
   (TM, "release"),
   (TM, "movearm", "home"),
   (AMR,"say", "finished"),
  ]

  for command in commands:
   response = self.process_request(*command)
   if response is None:
    self.get_logger().error("Aborting demo due to failed command")
    break

def main(args=None):
 rclpy.init(args=args)

 node = DemoRoutine()
 node.run_demo()

if __name__ == ’__main__’:
 main()
```