agriculture

MDPI

Article

KOALA: A Modular Dual-Arm Robot for Automated Precision Pruning Equipped with Cross-Functionality Sensor Fusion

Charan Vikram [1], Sidharth Jeyabal [1], Prithvi Krishna Chittoor [1,*], Sathian Pookkuttath [1], Mohan Rajesh Elara [1] and Wang You [2]

[1] Engineering Product Development Pillar, Singapore University of Technology and Design, Singapore 487372, Singapore; charanvikram050203@gmail.com (C.V.); sidharth_jeyabal@alumni.sutd.edu.sg (S.J.); sathian_pookkuttath@sutd.edu.sg (S.P.); rajeshelara@sutd.edu.sg (M.R.E.)

[2] College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China; king_wy@zju.edu.cn

* Correspondence: prithvi_chittoor@sutd.edu.sg

Abstract: Landscape maintenance is essential for ensuring agricultural productivity, promoting sustainable land use, and preserving soil and ecosystem health. Pruning is a labor-intensive task among landscaping applications that often involves repetitive pruning operations. To address these limitations, this paper presents the development of a dual-arm holonomic robot (called the KOALA robot) for precision plant pruning. The robot utilizes a cross-functionality sensor fusion approach, combining light detection and ranging (LiDAR) sensor and depth camera data for plant recognition and isolating the data points that require pruning. The You Only Look Once v8 (YOLOv8) object detection model powers the plant detection algorithm, achieving a 98.5% pruning plant detection rate and a 95% pruning accuracy using camera, depth sensor, and LiDAR data. The fused data allows the robot to identify the target boxwood plants, assess the density of the pruning area, and optimize the pruning path. The robot operates at a pruning speed of 10–50 cm/s and has a maximum robot travel speed of 0.5 m/s, with the ability to perform up to 4 h of pruning. The robot's base can lift 400 kg, ensuring stability and versatility for multiple applications. The findings demonstrate the robot's potential to significantly enhance efficiency, reduce labor requirements, and improve landscape maintenance precision compared to those of traditional manual methods. This paves the way for further advancements in automating repetitive tasks within landscaping applications.

Keywords: dual-arm robot; hedge shaping; landscaping; pruning; robot vision

check for updates

Citation: Vikram, C.; Jeyabal, S.; Chittoor, P.K.; Pookkuttath, S.; Elara, M.R.; You, W. KOALA: A Modular Dual-Arm Robot for Automated Precision Pruning Equipped with Cross-Functionality Sensor Fusion. *Agriculture* **2024**, *14*, 1852. https://doi.org/10.3390/agriculture14101852

Academic Editor: José Lima

Received: 21 September 2024
Revised: 18 October 2024
Accepted: 18 October 2024
Published: 21 October 2024

1. Introduction

Major cities are undergoing rapid development, with an influx of people from rural areas creating a significant demand for new housing. Consequently, many trees and plants have been removed to accommodate this urban expansion. To maintain adequate vegetation in urban areas [1], substantial efforts are made to plant trees and small shrubs within cities. However, if these shrubs are left to grow without regular pruning, they can become hazardous to pedestrians and detract from the aesthetic appeal of urban environments. Currently, pruning is performed manually, and this method is labor-intensive [2] and expensive. A decade ago, manual machines were used for pruning, requiring significant human participation. With rising temperatures and increasing labor costs, governments have made substantial efforts to transition these operations to teleoperated or autonomous systems. Robots, which are already utilized in precision agriculture and data logging, offer a promising solution. Autonomous robots come with built-in intelligence, enabling them to maintain a closed-loop behavior, including the data logging of pruning areas and timely scheduling of landscape maintenance.

Many researchers have attempted to develop diverse robots for landscaping and agricultural applications [3–5]. The research carried out by Wang et al. [6] focused on developing a 6 degrees of freedom (DoF) Stewart parallel elevation platform that improves the harvesting precision and velocity of a robotic arm used for plucking tea leaves. The platform had a maximum deviation angle of 10°, a lifting distance of 15 cm, and a load capacity of 60 kg. The results meet the automated agricultural operation objectives, showing accurate motion control with millimeter-level stability. Jun et al. [7] proposed an efficient tomato harvesting robot, combining deep learning-based tomato detection, 3D perception, and manipulation. Arad et al. [8] proposed a visual servo system in which manipulator motion planning is carried out to move to the picture center coordinates of the recognized fruit while maintaining a predefined location. Ling et al. [9] introduce a dual-arm cooperative tomato harvesting method using a binocular vision sensor, achieving over 96% tomato detection accuracy at 10 frames per second. The system enables real-time control for precise end-point positioning within 10 mm. SepúLveda et al. [10] proposed a dual-arm aubergine harvesting robot with efficient fruit detection and coordination. It enables simultaneous harvesting, single-arm operation, and collaborative behavior to address occlusions, demonstrating a 91.67% success rate, with an average cycle time of 26 s per fruit. Li et al. [11] introduced a multi-arm apple harvesting robot system, emphasizing precise perception and collaborative control. The dual-arm approach achieves a median locating error reduction of up to 44.43%, a 33.3% decrease in average cycle time compared to heuristic-based algorithms, and harvest success rates ranging from 71.28% to 80.45%, with cycle times averaging between 5.8 to 6.7 s. Xiong et al. [12] used a dual 3-DoF cartesian robot for strawberry harvesting. The robot picked two strawberries at a time from clusters using two robotic arms and an obstacle-separation algorithm, averaging 4.6 s for the two-arm mode. Mu et al. [13] developed an end-effector using bionic fingers equipped with fiber sensors for precise fruit detection and pressure sensors for damage control. A kiwifruit harvesting robot with two 6-DoF jointed manipulators and custom grippers was created; it picks a single fruit in an average of 4–5 s and features a 94.2% success rate. A lightweight robotic arm designed for kiwifruit pollination underwent rigorous simulation and testing, yielding an impressive 85% success rate and boasting an efficiency of 78 min/mu, outperforming traditional manual methods by a significant margin [14]. A low-cost system was created to evaluate planning, manipulation, and sensing in a contemporary flat-canopied apple orchard. The machine vision system was tested, and it could localize fruit in 1.5 s, on average. At an average picking time of 6.0 s per fruit, the 7-DoF harvesting system successfully harvested 127 of the 150 attempted fruits, resulting in an overall success rate of 84% [15]. An apple-harvesting robot recently created in China grasped the stem with a spoon-shaped end-effector and then chopped it with an electric blade. This device, which employed an eye-in-hand sensor configuration for fruit localization, was shown to achieve an average harvesting time of 15.4 s per apple [16]. The authors of Refs. [17,18] discuss techniques utilized with unmanned ground vehicles for crop monitoring and navigation around the fields. Kim et al. [19] proposed an agricultural robot, “Purdue Ag-Bot”, which efficiently uses point cloud data from Ouster LiDAR and image data from an Intel RealSense camera implemented on a mobile robot and a 7-DoF robot, achieving a less than 10% error in regards to crop heights and stalk diameter. Physical sampling was performed, with a greater than 80% success rate [20–22]. A bush-trimming robot was proposed by Kamandar et al. [23], which aimed at identifying the bush area for trimming and moving parallel to the bush. The developed robot used vision-based navigation and four revolute joints to perform trimming actions parallel to the floor surface. An automated boxwood trimming robotic arm was developed by Kaljaca et al. [24] and Marrewijk et al. [25], which performed trimming operations under a controlled setting using a motion capture system, markers, and reference rings. The authors used the concept of collecting the point cloud data of the plant and estimating the required target shape. The pruning action was performed sequentially to achieve the desired shape of the plant.

Most research on landscaping and agricultural robots has focused on the perception of target information, the automated recognition algorithm [26,27], and the detection of target positions [28,29]. Furthermore, few reports focus on using high DoF robotic arms to precisely manipulate target points. Existing research reports on the operation of robotic arms not only reflect the low operating efficiency and accuracy of the robotic arms but also show that the robotic arms used are large and have a complex structure, high weight, high inertia, and high manufacturing cost [10,30,31]. Furthermore, many robotic systems experience long cycle times or require complex coordination mechanisms, limiting their real-world applicability in dynamic environments. Although some robots succeed in specific tasks like harvesting, these systems often require more adaptability and efficiency for precision pruning tasks in urban landscapes. To address these limitations, this study proposes a dual-arm holonomic KOALA robot that aims to improve efficiency and precision through cross-functionality sensor fusion and collaborative control. The robot integrates LiDAR sensors, RGB cameras, and depth data, creating a highly detailed map of the environment for accurate pruning decisions. Based on the existing literature, very few studies report fusing LiDAR, RGB, and depth data for extracting precision pruning points, a gap that this research aims to address. This sensor data integration enables the system to identify target pruning points with high accuracy and to streamline the pruning process, and this method needs to be studied and experimentally tested for its feasibility. The main contributions of the paper include the following:

a. The design of a dual-arm, holonomic robot capable of navigating urban landscapes with high payload capacity and stable locomotion, offering enhanced versatility for pruning and related tasks.
b. The development of a precision pruning algorithm that fuses data from LiDAR, RGB, and depth sensors, allowing for the accurate detection of overgrown plants and the assessment of pruning areas.

2. Design Overview

The robot design is divided into four major sub-sections: robot chassis design, manipulator design, electrical connections, and kinematic analysis.

2.1. Robot Chassis Design

The KOALA robot's chassis design was driven by the need for high maneuverability, stability, and the ability to carry heavy payloads while performing precision pruning tasks in tightly confined spaces. The chassis is split into two primary subsystems, i.e., the locomotion system at the bottom and the manipulation system at the top, as seen in Figure 1. These subsystems were focused on meeting the design objectives of high maneuverability, stable load-bearing capacity, precision localization, and navigation in complex urban environments, while carrying all necessary equipment and sensors.

Two pairs of mecanum wheels were selected for the locomotion system to provide a zero-turning radius, allowing the robot to navigate confined and cluttered areas without requiring extra maneuvering space. Powered by a 48 V, 40 Ah LiFePO$_4$ battery and driven by high-torque Oriental motors, the locomotion system ensures long operational times (up to 4 h). It supports a payload capacity of 400 kg, including the two Kinova Gen3 robotic arms. The payload capacity of the chassis is critical, as it must support the Kinova Gen3 arms, which are used for precision pruning. Each arm has a maximum reach of 902 mm and can carry up to 4 kg at mid-range. The chassis provides the stability required for the arms to operate accurately, ensuring they can prune plants precisely, even when fully extended or manipulating complex angles. This design ensures that the robot is versatile enough to handle various pruning tasks. For accurate perception and localization, the KOALA robot contains two 2D-LiDARs, mounted diagonally, to provide a 360° view of its surroundings. In addition, a 3D-LiDAR works with the Intel RealSense D435i depth camera to generate a detailed overview of the pruning area. This sensor fusion allows the robot to identify and localize plants accurately, ensuring precise targeting during pruning.

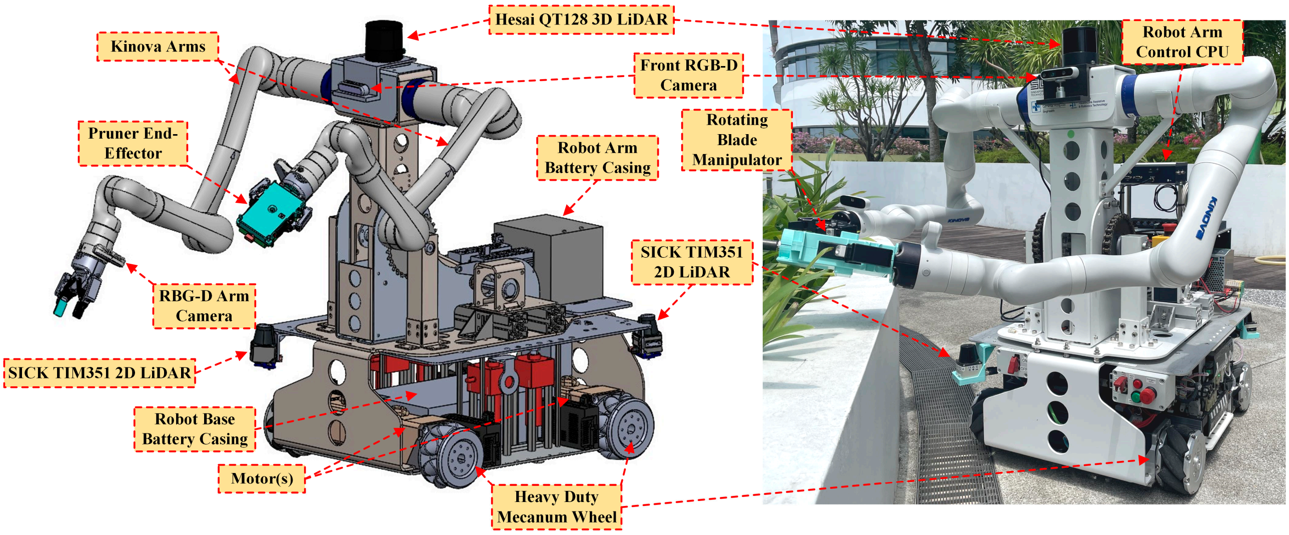


Figure 1. Robot chassis 3D model and the developed dual arm precision pruning robot.

Lastly, the robot features two industrial PCs (IPC) that handle real-time sensor data processing and control the robot's locomotion and arm manipulation. One PC is dedicated to processing the localization and mapping tasks, while the other handles the control of the manipulator arms. This computational setup ensures that the robot can quickly and accurately respond to environmental changes, while also enabling it to plan and execute precise pruning actions. The design of the robot concentrates on maneuverability, payload stability, and localization accuracy, all of which contribute to the robot's ability to perform efficient and accurate pruning in urban landscapes.

2.2. End-Effector Design

The overall structure of this 3D-printed end effector for the robotic arm consists of the 3D-printed housing, a DC motor, Li-ion batteries, the blade mount, and the blade, as shown in Figure 2. The main body of the end effector, known as the housing, is created from two 3D-printed halves (top and bottom casing) that are fastened together, providing a strong and lightweight structure to house the internal components. The end effector operates using an onboard power source, the Li-ion battery, and can be powered by the robot's battery. The rotating blade is the cutting element of the end effector, designed for pruning and intricate cutting. The rotating blades and the blade mount are mechanically coupled to the motor shaft.

In addition to its core safety features, the robotic arm incorporates a robust sensor failure mechanism. The integrated force and vibration sensors continuously monitor the end-effector's operation, ensuring that any anomalies, such as excessive force or abnormal vibrations, are quickly detected. If these sensors register values outside predefined safety thresholds, the system automatically initiates a shutdown by cutting off the power supply to the blade. This proactive safety mechanism prevents potential damage to the equipment and mitigates the risk of injury, ensuring that the robot operates safely and effectively in its collaborative environment with humans. In addition, the operator is provided with an emergency shut-off switch for immediate manual intervention, further enhancing the safety of the system.

To determine the required cutting force, experiments performed by Ref. [32] on apple tree branches are taken as a reference, which shows the cutting force for a 12–25 mm diameter branch varies from 4.58 to 18 N. In the current application of boxwood plant pruning, the selected plant consists of thin and soft stems that do not require significant force. For the current 12 V 775 DC motor, the no-load speed is 12,000 RPM, with a stall torque of 79 Ncm at 14.4 V (or 0.79 Nm), and the angular speed is calculated to be 1256.64 rad/s. Assuming that the full load torque is around 20% of the stall torque, thus, the full load torque is estimated to be 0.2 × 0.79 Nm = 0.158 Nm. Therefore, for a 15 cm blade, the cutting force of the blade is given by (full load torque/radius of the blade) = 0.158 Nm/ 0.075 m = 2.11 N.

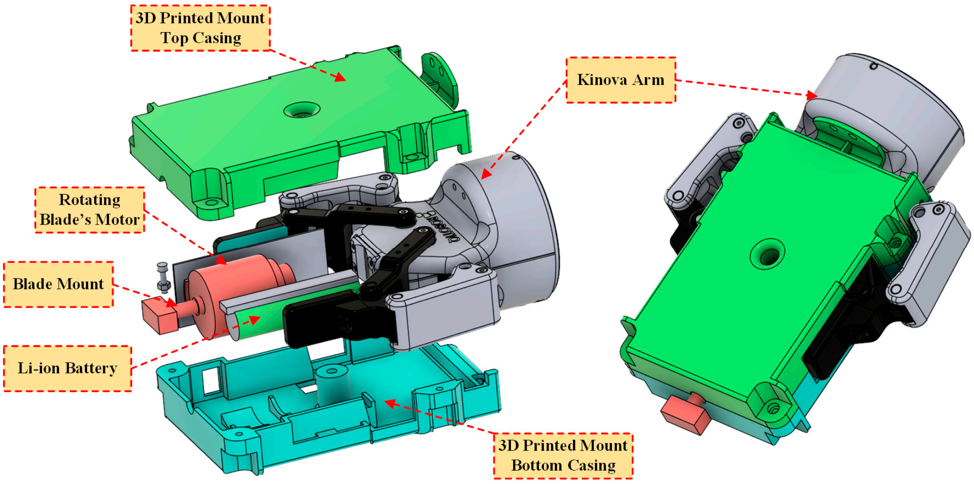


Figure 2. Detailed 3D model of the robot end-effector.

The calculations show that the stainless steel 15 cm diameter blade attached to the motor shaft generates a 2.11 N force, sufficient for the current pruning/trimming applications.

2.3. Electrical Design

The complete electrical system of the developed device is powered by two 48 V, 40 Ah batteries, as illustrated in Figure 3. These batteries provide the primary energy source for the entire system. Several voltage regulators are employed to convert the 48 V battery voltage to the lower levels (12 V and 5 V) required by various components, such as the industrial PC (IPC), motor drivers, network switches, and LiDAR sensors. Bus bars, which are conducting strips, are utilized to distribute power efficiently to multiple devices within the system, with separate bus bars designated for the 12 V and 5 V power levels. The Advantech IPC necessitates a 12 V power supply, provided by an SD-50C-12 voltage regulator and a 12 V bus bar. Network switches, including a TP-Link TL-SG105 and another switch, also require a 12 V power supply provided by an SD-15C-5 voltage regulator and a 5 V bus bar. Driver modules, such as the BLV640NM405, operate on the 48 V bus bar and are crucial for controlling motor functions. The system incorporates two SICK LiDAR sensors (TA1581-2050101), which demand a 12 V power supply sourced from an RSD-150C-12 voltage regulator and a 12 V bus bar. Notably, the top LiDAR sensor features a 360° horizontal field of view (FOV) and a 105.2° vertical FOV, with a maximum range of 50 m. Additionally, the system includes two Kinova Gen3 arms powered by the second 48 V, 40 Ah battery, ensuring robust performance and operational flexibility. The detailed specifications are presented in Table 1.

The operating time of the robotic arm control system can be calculated based on the power consumption of all components and the capacity of the installed 48 V, 40 Ah $LiFePO_4$ battery. Each Kinova arm consumes 155 W, while the industrial PC uses 20 W, the network switch consumes 2.41 W, the 3D LiDAR uses 30 W, the Intel RealSense camera draws 1.5 W, and the pruning DC motor draws 16.8 W. This brings the total power consumption to 380.71 W. With the battery providing 1536 Wh of usable energy (80% of the full capacity due to a 20% safety margin), the operating time of the entire system is calculated by dividing the battery's available energy by the total power consumption. Thus, ideally, the system can operate for approximately 4.03 h before depleting the battery. In practice, the system has been observed to perform optimally for 2.5~3 h. Similarly, the total operational time of the robot base at full load, considering four motors (each rated at 400 W) moving the robot at 0.5 m/s, the IPC consuming 20 W, the network switch consuming 2.41 W, and two LiDARs consuming 4 W each, is approximately 0.94 h (or about 56.5 min), when powered by a 48 V, 40 Ah battery with a 20% safety margin. In practice, the robot is not always moving; it makes minor corrections during pruning for a given area. Thus, the base does not operate at full-load at all times, pushing the usable time of the robot to 4–5 h.

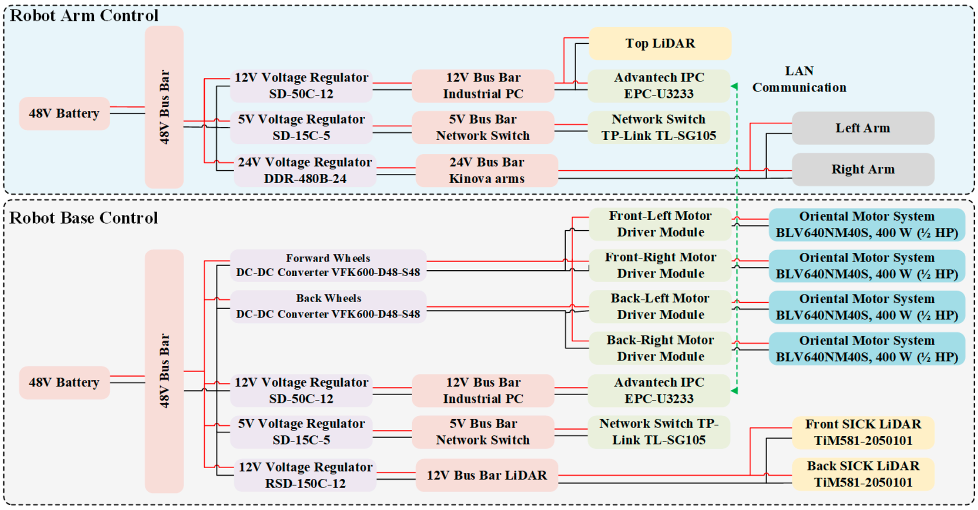

Robot Arm Control
48V Battery
48V Bus Bar
12V Voltage Regulator SD-50C-12
5V Voltage Regulator SD-15C-5
24V Voltage Regulator DDR-480B-24
12V Bus Bar Industrial PC
5V Bus Bar Network Switch
24V Bus Bar Kinova arms
Top LiDAR
Advantech IPC EPC-U3233
Network Switch TP-Link TL-SG105
LAN Communication
Left Arm
Right Arm
Robot Base Control
48V Battery
48V Bus Bar
Forward Wheels DC-DC Converter VFK600-D48-S48
Back Wheels DC-DC Converter VFK600-D48-S48
12V Voltage Regulator SD-50C-12
5V Voltage Regulator SD-15C-5
12V Voltage Regulator RSD-150C-12
12V Bus Bar Industrial PC
5V Bus Bar Network Switch
12V Bus Bar LiDAR
Front-Left Motor Driver Module
Front-Right Motor Driver Module
Back-Left Motor Driver Module
Back-Right Motor Driver Module
Advantech IPC EPC-U3233
Network Switch TP-Link TL-SG105
Oriental Motor System BLV640NM40S, 400 W (½ HP)
Oriental Motor System BLV640NM40S, 400 W (½ HP)
Oriental Motor System BLV640NM40S, 400 W (½ HP)
Oriental Motor System BLV640NM40S, 400 W (½ HP)
Front SICK LiDAR TiM581-2050101
Back SICK LiDAR TiM581-2050101

Figure 3. Block diagram of the electrical subsystem of the KOALA robot.

Table 1. Specifications of the components in the KOALA robot.

Product	Specification	Product	Specification
Battery	48 V, 40 Ah, $LiFePO_4$	Network switch	5-port RJ45 LAN
2D LiDAR	TiM581-2050101, range 0.05~25 m	IPC	EPC-U3233, Intel core i7-8665UE, Ubuntu
3D LiDAR	Hesai QT128, 105.2° vertical FOV, range 0.05~50 m, ±3 cm accuracy	DC-DC converter	VFK 600-D48-S48 input: 36~75 V, output: 48 V, 12.5 A, 600 W
Motor system	BLV640NM40S, 400 W Electromagnetic brake 48 V, 11 A (rated)	Kinova arms	Gen3, 7 DoF, 4 kg payload, 902 mm max. reach, max. cartesian translation speed is 50 cm/s
Voltage regulators	5 V regulator: SD-15C-5 12V regulator: SD-50C-12	Inertial measurement unit (IMU)	Vectornav VN-100

2.4. Kinematic Analysis

The 7-DoF allows access to hard-to-reach areas during the pruning operation. The joint angles for axes 1, 2, 3, 4, 5, 6, and 7 are in ranges of −inf (infinity) to +inf, −128.9° to +128.9°, −inf to +inf, −147.8° to +147.8°, −inf to +inf, −120.3° to +120.3°, and −inf and +inf, respectively. The Kinova robot has a fixed frame, namely a base frame. This frame must be transformed to the moving frame tool by finding the homogenous transform (H) from base to link1, link1 to link2, link2 to link3, link3 to link4, link4 to link5, link5 to link6, and link7 to tool. All the homogenous transform matrices obtained from the base to the tool frame are multiplied to get the transform from the base frame to the tool frame. Equation (1) obtains the forward kinematics of the 7-DoF Kinova arm. Below is the homogenous transform from the previous frame to the current frame H_i^{i-1}, where i is the current link; since this is a 7-DoF robot, the maximum is 7.

$$H_{\text{base to tool}} = {}^{b}_{1}H \times {}^{1}_{2}H \times {}^{2}_{3}H \times {}^{3}_{4}H \times {}^{5}_{6}H \times {}^{7}_{\text{tool}}H \quad (1)$$

$$ {}^{b}_{1}\mathrm{H} = \begin{bmatrix} \mathrm{cq}_1 & -\mathrm{sq}_2 & 0 & 0 \\ -\mathrm{sq}_1 & -\mathrm{cq}_1 & 0 & 0 \\ 0 & 0 & -1 & 0.1564 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{1}_{2}\mathrm{H} = \begin{bmatrix} \mathrm{cq}_2 & -\mathrm{sq}_2 & 0 & 0 \\ 0 & 0 & -1 & 0.0054 \\ \mathrm{sq}_2 & \mathrm{cq}_2 & 0 & -0.1284 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{2}_{3}\mathrm{H} = \begin{bmatrix} \mathrm{cq}_3 & -\mathrm{sq}_3 & 0 & 0 \\ 0 & 0 & 1 & -0.2104 \\ -\mathrm{sq}_3 & -\mathrm{cq}_3 & 0 & -0.0064 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{3}_{4}\mathrm{H} = \begin{bmatrix} \mathrm{cq}_4 & -\mathrm{sq}_4 & 0 & 0 \\ 0 & 0 & -1 & -0.0064 \\ \mathrm{sq}_4 & \mathrm{cq}_4 & 0 & -0.1284 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{4}_{5}\mathrm{H} = \begin{bmatrix} \mathrm{cq}_5 & -\mathrm{sq}_5 & 0 & 0 \\ 0 & 0 & 1 & -0.2084 \\ -\mathrm{sq}_5 & -\mathrm{cq}_5 & 0 & -0.0064 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{5}_{6}\mathrm{H} = \begin{bmatrix} \mathrm{cq}_6 & -\mathrm{sq}_6 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \mathrm{sq}_6 & \mathrm{cq}_6 & 0 & -0.1059 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{6}_{7}\mathrm{H} = \begin{bmatrix} \mathrm{cq}_7 & -\mathrm{sq}_7 & 0 & 0 \\ 0 & 0 & 1 & -0.1059 \\ -\mathrm{sq}_7 & -\mathrm{cq}_7 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

where $\mathrm{sq_n}$ and $\mathrm{cq_n}$ represents $\sin(\mathrm{q_n})$ and $\cos(\mathrm{q_n})$, respectively.

To reach point *X*, Y, and Z in space, an inverse kinematic transform must be performed, followed by the forward kinematic transform, and equating to the transform matrix H. The target point in 3D space is determined by the LiDAR and the depth-camera integration algorithm, which is explained in detail in the following section. This transformation is fed to the robot arm controller, as shown in Equation (2).

$$ \mathrm{H} = \begin{bmatrix} n_x & m_x & o_x & \mathrm{X} \\ n_y & n_y & n_y & \mathrm{Y} \\ n_z & n_z & n_z & \mathrm{Z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

where n_x, m_x, o_x, n_y, m_y, o_y, n_z, m_z, and o_z represent the components of the rotation matrix part of the transformation, and the tool coordinate in 3D space relative to the base frame is given by the following:

$$ \begin{aligned} X = {} & 0.3143 \cdot cq_1 \cdot cq_4 \cdot sq_2 + 0.3388 \cdot cq_1 \cdot sq_2 - 0.1059 \cdot cq_6 \cdot (-cq_1 \cdot cq_4 \cdot sq_2 + sq_4 \cdot (-cq_1 \cdot cq_2 \cdot cq_3 + sq_1 \cdot sq_3)) \\ & -0.0118 \cdot sq_1 - 0.3143 \cdot sq_4 \cdot (-cq_1 \cdot cq_2 \cdot cq_3 + sq_1 \cdot sq_3) - 0.1059 \cdot sq_6 \\ & \cdot (cq_5 \cdot (cq_1 \cdot sq_2 \cdot sq_4 + cq_4 \cdot (-cq_1 \cdot cq_2 \cdot cq_3 + sq_1 \cdot sq_3)) + sq_5 \cdot (cq_1 \cdot cq_2 \cdot sq_3 + cq_3 \cdot sq_1)) \end{aligned} $$

$$ \begin{aligned} Y = {} & -0.0118 \cdot cq_1 - 0.3143 \cdot cq_4 \cdot sq_1 \cdot sq_2 - 0.1059 \cdot cq_6 \cdot (cq_4 \cdot sq_1 \cdot sq_2 + sq_4 \cdot (cq_1 \cdot sq_3 + cq_2 \cdot cq_3 \cdot sq_1)) - 0.3388 \\ & \cdot sq_1 \cdot sq_2 - 0.3143 \cdot sq_4 \cdot (cq_1 \cdot sq_3 + cq_2 \cdot cq_3 \cdot sq_1) - 0.1059 \cdot sq_6 \\ & \cdot (cq_5 \cdot (cq_4 \cdot (cq_1 \cdot sq_3 + cq_2 \cdot cq_3 \cdot sq_1) - sq_1 \cdot sq_2 \cdot sq_4) - sq_5 \cdot (-cq_1 \cdot cq_3 + cq_2 \cdot sq_1 \cdot sq_3)) \end{aligned} $$

$$ \begin{aligned} Z = {} & 0.3143 \cdot cq_2 \cdot cq_4 + 0.3388 \cdot cq_2 - 0.3143 \cdot cq_3 \cdot sq_2 \cdot sq_4 - 0.1059 \\ & \cdot cq_6 \cdot (-cq_2 \cdot cq_4 + cq_3 \cdot sq_2 \cdot sq_4) - 0.1059 \cdot sq_6 \cdot (cq_5 \cdot (cq_2 \cdot sq_4 + cq_3 \cdot cq_4 \cdot sq_2) - sq_2 \cdot sq_3 \cdot sq_5) + 0.2848 \end{aligned} $$

3. Methodology for Mapping, Precision Pruning, and Navigation

The KOALA robot utilizes a two-phase operational methodology that consists of mapping and pruning, as illustrated in Figure 4. The process begins with a teleoperated mapping phase, where the robot is remotely navigated through the designated deployment zone to construct a comprehensive 2D map of the environment. This mapping is crucial for identifying key pruning areas. The process starts by assigning waypoints near the pruning location. The robot uses 2D-LiDARs to collect a 360° scan of the environment. Both LiDARs operate on the same scanning plane, ensuring consistency in the data collected. Each LiDAR provides a 270° scan (illustrated by the shaded regions in Figure 5), with one LiDAR positioned at the front of the robot and the other at the back. When the LiDAR scans are merged, the resultant scan appears as though it is projected from the center of the robot, providing a seamless 360° view of the robot's surroundings. This scan is merged with odometry and IMU data to localize the robot's position accurately. The robot also initializes its RGB-D camera to detect the areas of green shrubs, which are marked as hotspots for pruning.

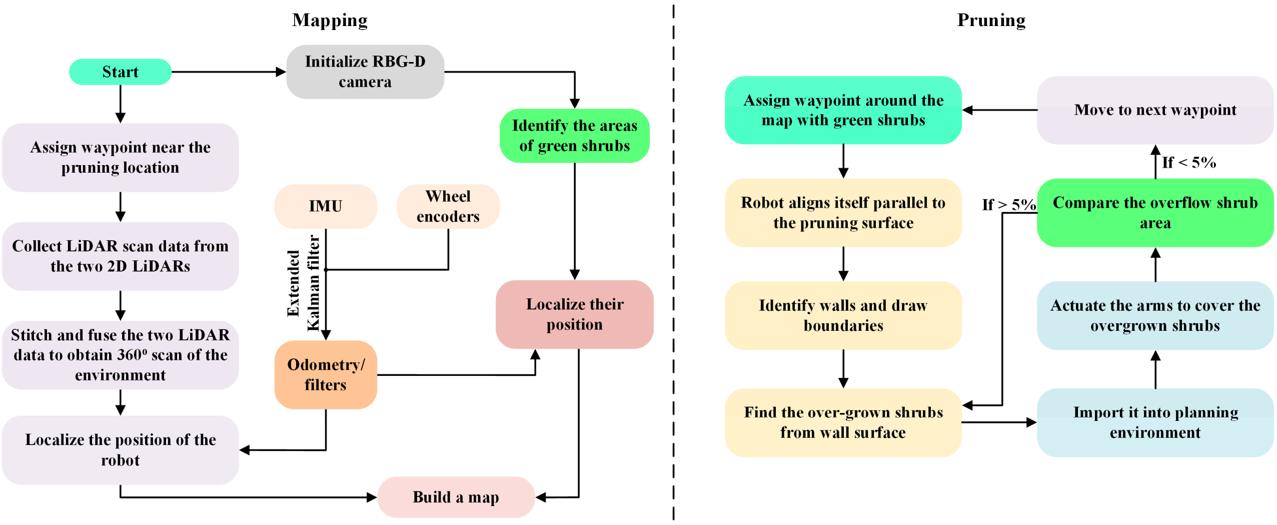

Mapping
Start
Initialize RBG-D camera
Identify the areas of green shrubs
Assign waypoint near the pruning location
IMU
Wheel encoders
Collect LiDAR scan data from the two 2D LiDARs
Extended Kalman filter
Localize their position
Stitch and fuse the two LiDAR data to obtain 360° scan of the environment
Odometry/ filters
Localize the position of the robot
Build a map
Pruning
Assign waypoint around the map with green shrubs
Move to next waypoint
If < 5%
Robot aligns itself parallel to the pruning surface
If > 5%
Compare the overflow shrub area
Identify walls and draw boundaries
Actuate the arms to cover the overgrown shrubs
Find the over-grown shrubs from wall surface
Import it into planning environment

Figure 4. Methodology of the mapping and pruning process.

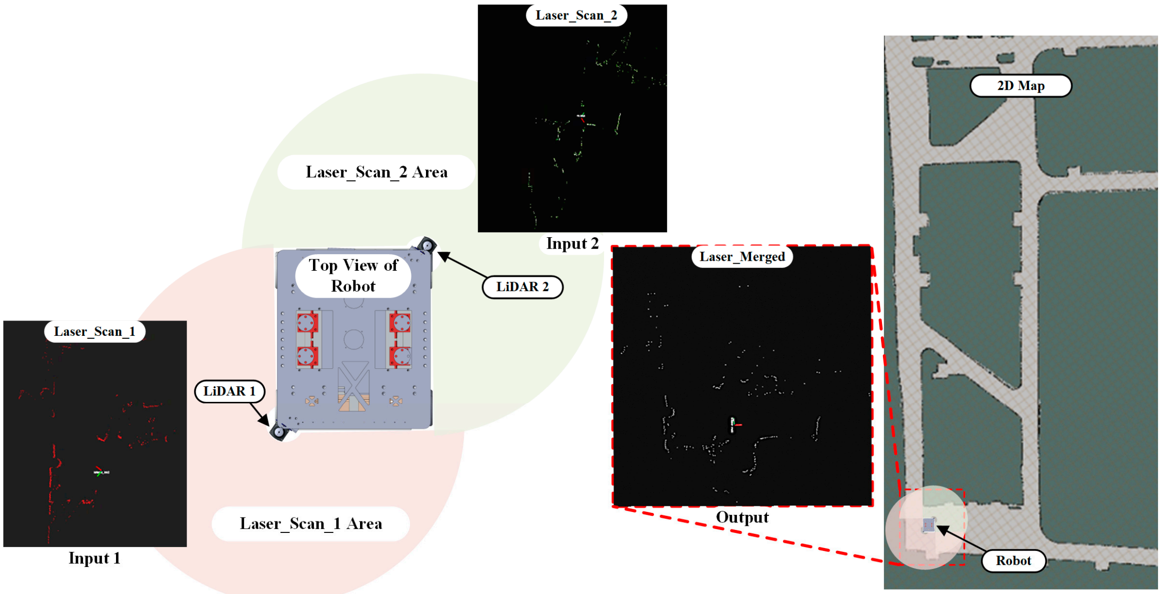


Figure 5. Output of the 2D LiDARs merged to give 360° view around the robot.

Once the mapping is complete and the hotspots are identified, the robot transitions to the autonomous pruning phase. In this phase, the waypoints around the green shrub areas are assigned to the robot (as shown in the right section of Figure 4). The robot aligns itself parallel to the pruning surface, efficiently covering the overgrown shrubs. The system then identifies the walls and draws boundaries around the plants, helping it distinguish overgrown shrubs from wall surfaces. The robot begins pruning the identified areas using its dual Kinova Gen3 manipulators with the custom-designed trimming tool.

The KOALA robot compares the overgrown shrub area and systematically reduces the number of prune points by actuating its arms to cover the targeted shrub regions. Once the number of pruning points falls below a predefined threshold (5%), the robot moves to the next waypoint. This cyclic process ensures continuous and efficient maintenance of the designated areas, reducing the overgrown shrubs in a controlled manner while maintaining operational efficiency throughout the pruning process. The mapping and pruning process shown in Figure 4 highlights the step-by-step methodology, ensuring clarity in both the map-building and pruning phases.

3.1. Remote Robotic Mapping and Annotation

The KOALA robot is teleoperated around the deployment zone, utilizing the Gmapping[] array SLAM (simultaneous localization and mapping) algorithm to construct a map from the merged LiDAR scans discussed in the previous section. It uses the robot's IMU data and the LiDAR scan data to accurately build a map of the environment while simultaneously keeping track of the robot's location within it. The IMU provides crucial information about the robot's orientation and movement, enhancing the accuracy of the mapping process by compensating for any discrepancies in the scan data due to motion. Adaptive Monte Carlo localization (AMCL) is also used in conjunction with the SLAM algorithm to refine the robot's position on the generated map. AMCL applies a probabilistic approach to localize the robot by comparing the real-time LiDAR scan data with the existing map, allowing for precise navigation and operation within the deployment zone. The map taken from the deployment zone is illustrated in Figure 6.

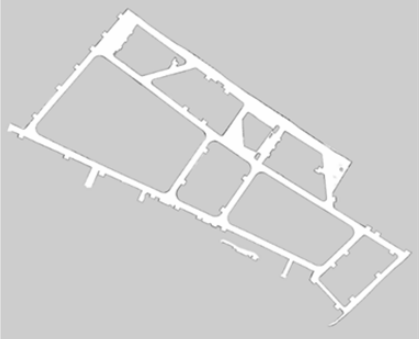


Figure 6. A 2D map of the deployment zone retrieved from the 2D LiDARs and IMUs.

An Intel-RealSense depth camera is mounted on top of the robot to capture a comprehensive view of the environment in which the robot operates. The YOLOv8 (You Only Look Once version 8) [33] model has been trained with a boxwood dataset to enhance the robot's ability to detect plants accurately. The dataset was divided into training, valid, test sets for validation, and evaluating detection parameters. In this context, YOLOv8 has been specifically employed to detect and segment plants within the robot's operational environment. The model's ability to provide precise masks for plant detection allows the robot to accurately distinguish plants from other objects and background elements.

The masks generated by YOLOv8 are used to annotate the map created by the SLAM process. Pixels corresponding to the highlighted areas in the masks are subject to distortion. To correct this distortion, the inverse Brown–Conrady model is applied, as illustrated in Equation (3). The inverse Brown–Conrady model is a mathematical approach used to correct lens distortion by mapping distorted points back to their undistorted locations.

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} - \begin{bmatrix} c_x \\ c_y \end{bmatrix} - \begin{bmatrix} u_i\left(k_1 r_i^2 + k_2 r_i^4 + k_3 r_i^6\right) \\ y_i\left(k_1 r_i^2 + k_2 r_i^4 + k_3 r_i^6\right) \end{bmatrix} - \begin{bmatrix} p_1\left(r_i^2 + 2u_i^2\right) + 2p_2 u_i v_i \\ p_2\left(r_i^2 + 2v_i^2\right) + 2p_1 u_i v_i \end{bmatrix} = f \cdot \begin{bmatrix} \frac{X_c}{Z_c} \\ \frac{Y_c}{Z_c} \end{bmatrix} \quad (3)$$

Distortion coefficients = k_1, k_2, k_3, p_1, p_2;
r_i = the distance from the principal point of our camera in image space, relative to the principal point;
x_i, y_i = coordinates in distorted pixel space;
X_c, Y_c = coordinates in corrected pixel space.

Once the points are undistorted using this model, the pinhole camera inverse projection method, as shown in Equation (4), is employed. This method projects the undistorted

2D points back into the 3D camera coordinate system. After obtaining these points relative to the camera, they are marked on the map as locations where pruning operations must take place, as shown in Figure 7.

$$\Gamma_{\text{object}}^{\text{camera}\,-1} \cdot \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix} \qquad \Gamma_{\text{object}}^{\text{camera}} = \text{Intrinsic Matrix} \tag{4}$$

These marked points in the map are the specific locations the KOALA robot will visit for pruning. This mapping and annotation process is performed only once for a particular deployment zone, ensuring that the robot efficiently navigates and performs tasks in the designated areas.

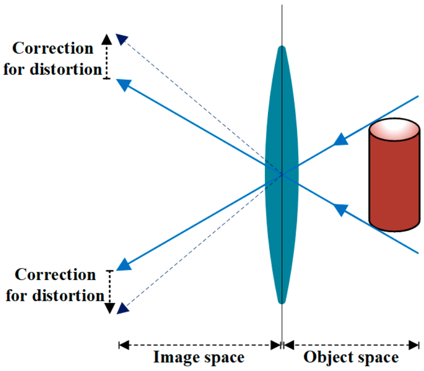


Figure 7. Distortion correction for 2D points projected into the 3D camera coordinate system.

3.2. Segmentation and Identifying Pruning Points

Once the annotated map is provided, the KOALA robot will navigate to the designated positions, identify the overgrown leaves, and prune them using its dual-arm capability. The flow of the pruning processes is illustrated in Figure 4.

3.2.1. Wall Segmentation

The Hesai QT-128 LiDAR is employed to identify overgrown points. This LiDAR features 128 channels, with a vertical field of view (FOV) of 105.2° and a horizontal FOV of 360°. The data acquired from the 3D LiDAR is pre-processed to reduce the number of points, thereby decreasing the processing time required for identifying overgrown points.

During preprocessing, the point cloud is divided into 3D boxes, and the points within each box are approximated by their centroid. These 3D boxes have dimensions of 0.03 × 0.03 × 0.03 m. Initially, the input point cloud contained 115,200 points, which were reduced to 51,573 points after preprocessing. The point clouds before and after filtering are illustrated in Figure 8.

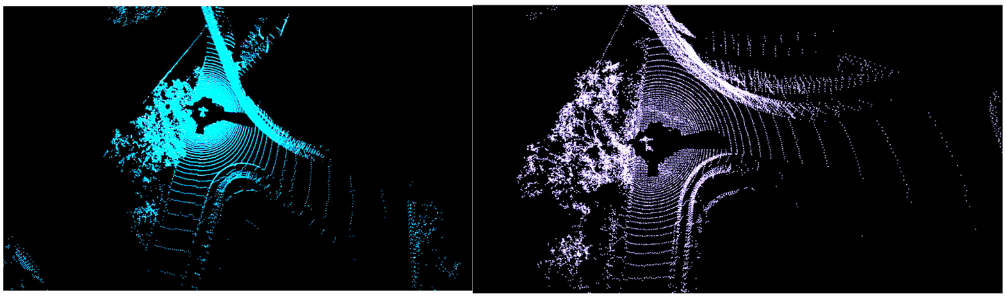


Figure 8. LiDAR point cloud preprocessing.

After preprocessing, a vertical plane parallel to the 3D LiDAR is identified using a method based on the random sample consensus algorithm. Initially, three points are randomly selected from the point cloud and treated as inliers to define a candidate plane. As described in Algorithm 1, the algorithm checks if additional nearby points lie on this candidate plane. If nearby points are found to fit the plane within a specified distance threshold, these points are added as inliers. This process of selecting three random points and expanding the set of inliers is repeated iteratively until the maximum number of iterations is reached.

Algorithm 1: Wall detection algorithm for identifying overgrowth

```
Input: Point Cloud $PC$
Output: Dominant Plane [a, b, c, d] $\rightarrow$ $aX + bY + cZ + d = 0$
Initialize:
$N$              ← Number of points in point cloud $PC$
$max\_iter$      ← Maximum number of iterations
$\epsilon$       ← Threshold value to determine if a data point fits well to the model (a 3D plane)
$min\_good\_fit$ ← Minimum number of nearby data points required to assert that a model fits well to data
$iter$           ← 0 // Current Iteration
$best\_err$      ← ∞ // Initial large error value
dominant plane ← [0, 0, 0, 0]
While iter < max_iter do:
    maybeInliers ← 3 randomly selected points from $PC$
    maybePlane ← a 3D plane fitted to maybeInliers
    alsoInliers ← [ ]
    For each point $p$ in $PC$ not in maybeInliers do:
        If point_to_model_distance ($p$, maybePlane) < $\epsilon$ then:
            Add $p$ to alsoInliers
        End If
    End For
    If number of points in alsoInliers>min_good_fit then
        betterPlane← A better 3D plane fitted to all points in maybeInliers and also Inliers (solving
        a least square problem)
        err← sum of distances from the points in maybeInliers and alsoInliers to betterPlane
        If err < best_err then
            best_err←err
            dominant_plane←betterPlane
        End If
    End If
    iter←iter+1
End While
Return: dominant_plane
```

This study set the maximum number of iterations to 10,000, and the distance threshold for checking point proximity was set to 0.01 m. The distance threshold (ϵ) was chosen based on the fixed deployment zone and the known environmental conditions. Since the robot operates in a controlled area, a pre-tuned fixed ϵ was used, which was experimentally found to provide an optimal balance between accuracy and computational efficiency in this specific environment. After completing the iterations, a voting scheme is employed to determine the plane with the maximum number of inliers. Additionally, the angle between the identified plane and the vertical axis of the LiDAR is examined. The plane with the highest number of inliers and an angle closest to zero degrees relative to the LiDAR's vertical axis is identified as the wall. This robust identification method ensures that the detected wall is both densely populated with data points and correctly aligned with the expected vertical orientation of the environment. The wall which was extracted from Figure 8 is illustrated in Figure 9.

After identifying the planar points, the normal distance between the plane and points is calculated as Equation (5). The points with a distance of less than 0.2 are selected as the overflow points. The overflow points are illustrated in green in Figure 10.

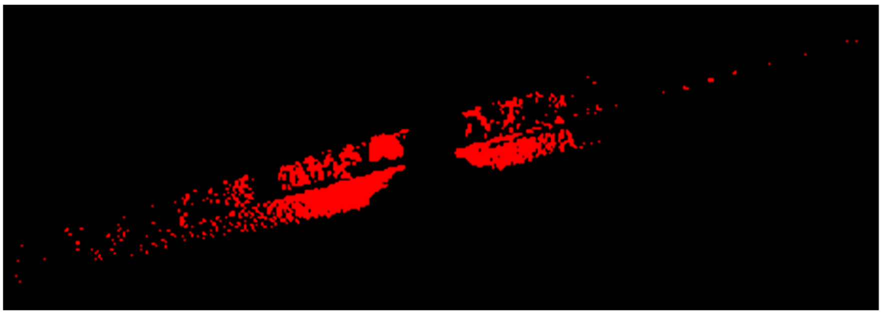


Figure 9. Wall segmentation from the LiDAR scan.

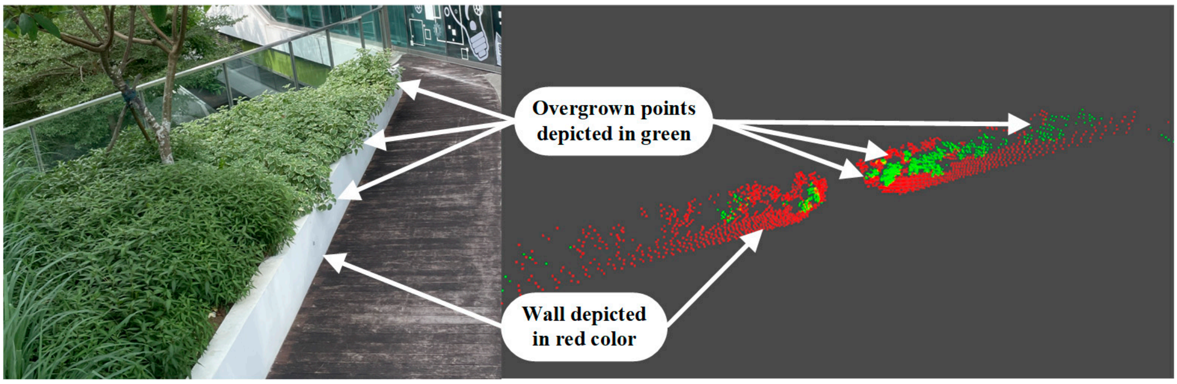

Overgrown points depicted in green
Wall depicted in red color

Figure 10. LiDAR point clouds preprocessing.

$$D = \frac{ax_1 + by_1 + cz_1 + d}{\sqrt{a^2 + b^2 + c^2}} \quad (5)$$

D → distance between plane and point;
$[a, b, c, d]$→ planar coefficients;
$[x_1, y_1, z_1]$ → point coordinates.

3.2.2. Plant Segmentation

The Intel-RealSense camera is employed to identify plant points from the overflow points. In this study, the experimental setup focuses exclusively on green plants. To achieve this, a mask is created to isolate green-colored areas within the images captured by the camera. This is achieved by defining a specific range of values in the HSV (hue, saturation, value) color space corresponding to various shades of green. HSV masking is a technique used to segment images based on color. The HSV color space is particularly useful for color segmentation because it separates chromatic content (hue) from intensity (value), making it easier to isolate specific colors under varying lighting conditions. A binary mask is generated by selecting appropriate lower and upper bounds for the hue, saturation, and value components. This mask highlights the regions of the image that fall within the specified HSV range, effectively identifying the green plants. This method ensures the accurate detection of plant points, facilitating further processing and analysis. The mask of the deployment zone received is shown in Figure 11.

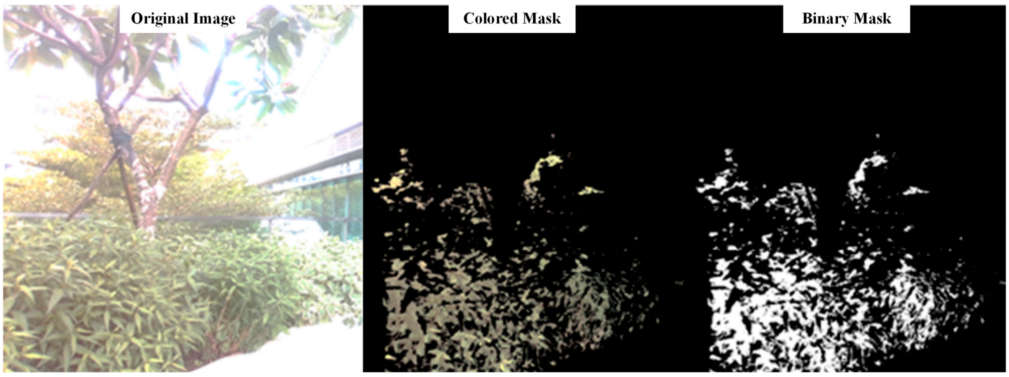


Figure 11. Segmentation of target area using a colored mask and a binary mask.

3.2.3. Converting Pruning Points into Point Cloud Data

As discussed at the beginning of Section 3.1, the points highlighted in the mask are undistorted using the inverse Brown–Conrady transformation. Subsequently, the real-world camera coordinates are determined using the pinhole camera model, as described in Equation (6). A binary AND operation is performed between the depth image and the binary color mask to ascertain the real-world depth. Points with a depth greater than 500 cm are filtered out. The depth values of the corresponding pixels are then multiplied by the depth scale factor (0.001 m for the RealSense camera) to obtain the real-time depth measurements. These points are then transformed from the camera frame to the LiDAR frame using the transformation equation provided in Equation (6). This process generates point clouds representing the identified plant areas in the LiDAR coordinate system.

$$
\begin{gathered}
\mathrm{T}\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \Gamma^{\text{camera}}_{\text{object}} \cdot \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix} r_i = \sqrt{x_i^2 + y_i^2} \\
\begin{bmatrix} x_i \\ y_i \end{bmatrix} - \begin{bmatrix} c_x \\ c_y \end{bmatrix} - \begin{bmatrix} u_i\left(k_1 r_i^2 + k_2 r_i^4 + k_3 r_i^6\right) \\ y_i\left(k_1 r_i^2 + k_2 r_i^4 + k_3 r_i^6\right) \end{bmatrix} - \begin{bmatrix} p_1\left(r_i^2 + 2u_i^2\right) + 2p_2 u_i v_i \\ p_2\left(r_i^2 + 2v_i^2\right) + 2p_1 u_i v_i \end{bmatrix} = f \cdot \begin{bmatrix} \frac{X_c}{Z_c} \\ \frac{Y_c}{Z_c} \end{bmatrix} \\
\text{Distortion Coefficients} : k_1, k_2, k_3, p_1, p_2
\end{gathered}
\tag{6}
$$

The projected plant points obtained from the Intel RealSense D435i camera and the overflow points identified from the 3D LiDAR are intersected. Points that belong to both sets are labeled as overflow plant points, indicating the regions that require pruning. These overflow plant points are then clustered based on their spatial proximity. To perform the pruning, the mean position of each cluster of overflow plant points is calculated. These mean positions are used as the target coordinates for the robotic arms. This allows for the cross-functionality sensor fusion of the RGB, depth, and LiDAR data, reducing noise in the required data. The robotic arms are then directed to these coordinates to execute the pruning operation. This step-by-step process ensures that the identified overflow plant points are accurately targeted and pruned, maintaining the health and growth of the plants in the deployment zone.

3.3. Robotic Waypoint Navigation and Manipulation

In the map-trimming process, segments are allocated to various annotated areas. These segments are defined based on the identified overflow plant points and are provided as waypoints for the robot. Each waypoint corresponds to a specific location the robot must visit to perform the pruning operation. The allocation of segments ensures that the robot systematically covers all designated areas, optimizing the pruning process. By following these waypoints, the robotic arms can precisely navigate to the targeted points, execute the necessary trimming tasks, and ensure comprehensive maintenance of the deployment zone. This systematic approach facilitates efficient and accurate pruning, maintaining the health and growth of the plants.

Trajectory planning is based on the overflow points described in previous sections. Firstly, for an allocated region, the system identifies a centroid, and trajectory is planned for the tool pose to move to that location. For each tool pose, various inverse kinematic (IK) solutions are generated with the IKFast algorithm and using a generalized traveling salesman problem (GTSP) over a graph whose nodes are the IK solutions for each target tool pose and whose edges are weighted with an objective function [34]. Once the gripper moves to that particular position, it performs a default trimming action toward the point where the density of overflow points is higher. This repetitive process of finding the mean of the points and performing trimming actions until the overflow points are less than 5% in the camera view continues.

The planned trajectory and the trimmer's ON/OFF signal were sent to an action server on top of the standard Kinova robot operating system driver. Initially, the action server generated a time-sampled version of the planned trajectory using quintic splines to ensure smooth motion, incorporating maximum speed and acceleration constraints. This time-sampled trajectory was then used to calculate appropriate joint speeds with a proportional-integral differential (PID) controller. According to the time-sampled trajectory, the PID controller compared the actual arm and the expected configurations at each instant. The Kinova manipulator then executed this trajectory segment by segment.

4. Results and Discussions

The experiment was conducted on Level 6 of Building 2 at the Singapore University of Technology and Design. After completing the mapping and annotation for a deployment zone, waypoints were assigned for the robot to navigate the zone and target the green areas. This experiment discusses three test cases, considering scenarios with 100%, 50%, and 5% of the identified overflow points.

4.1. Training, Annotation, and Pruning Results

As discussed in Section 3.1, the robot was teleoperated around the test level to perform mapping and annotate areas where plants are present. These areas were identified using the trained data of the YOLOv8 algorithm.

4.1.1. YOLOv8 Training

For the YOLOv8 training process, a dataset of 1000 annotated images of boxwood plants was carefully chosen, reflecting the specific target of the KOALA robot's pruning tasks. The dataset was divided into 70% for training, 20% for validation, and 10% for testing, allowing for robust model evaluation and parameter optimization. The model training was conducted using a Tesla T4 GPU, known for its high-performance capability in deep learning applications. This setup enabled efficient dataset processing, as YOLOv8's architecture requires significant computational resources due to its complex convolutional layers and large number of parameters.

The training objective was to maximize the accuracy of both the bounding box detection and mask segmentation of the boxwood plants, which is essential for real-time pruning. During training, the model's performance was tracked using several loss functions: box loss, object loss, classification loss, and corresponding validation losses. As shown in Figure 12, these losses decreased steadily as the training progressed, with significant reductions observed in the early epochs, indicating rapid learning.

Specifically, the training box loss and validation box loss dropped consistently, reflecting the model's improving ability to predict accurate bounding boxes around the plants. The training object loss and classification loss similarly showed a marked decline, highlighting the enhanced recognition of plant objects and the improved classification of the pruning targets within the images. The convergence of training and validation losses across all metrics indicates that the model was not overfitting, and that it generalizes well to unseen data, which is critical for its deployment in diverse real-world environments.

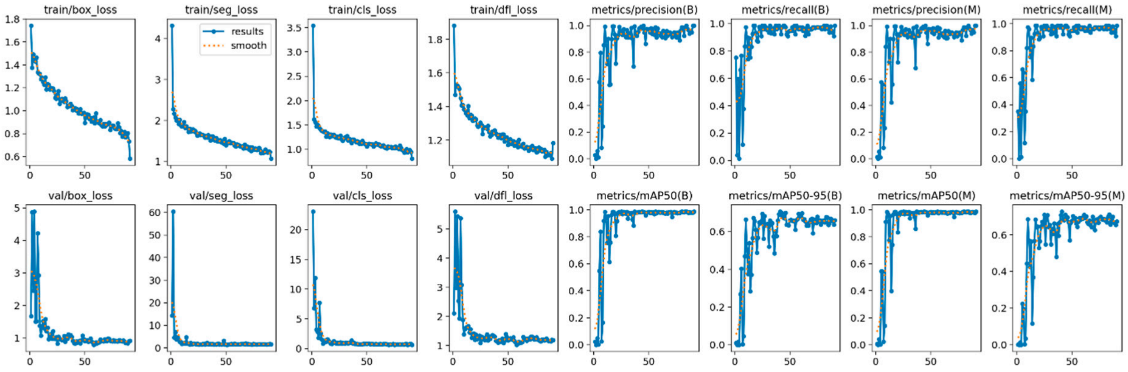


Figure 12. Training results of the YOLOv8 algorithm.

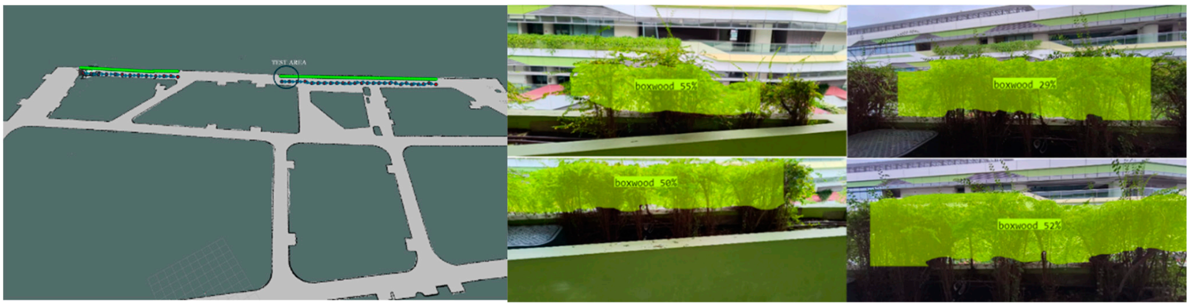
By the 41st epoch, the model reached its peak performance, achieving a mean average precision (mAP) of 0.985 at an IoU threshold of 50% (mAP 50) for bounding box detection. Additionally, the model exhibited a mAP of 0.706 across IoU thresholds from 50% to 95% (mAP 50-95), underscoring its robustness in accurately detecting boxwood plants, even under varying overlap conditions between predicted and actual bounding boxes.

The YOLOv8 model achieved similarly strong results for mask segmentation, with a mAP 50 of 0.985 and a mAP 50-95 of 0.727. This level of accuracy ensures precise plant segmentation, allowing the KOALA robot to accurately distinguish between plant areas and the background, even in cluttered or visually complex environments.

The evaluation metrics shown in the image provide further insight into the model's performance. Metrics such as precision, recall, and F1-score are plotted, showcasing their consistent improvement over time. Precision values near 1.0 indicate that the model has a low false-positive rate, while high recall values reflect its capability to detect nearly all relevant objects. The F1-score balances precision and recall and confirms the model's suitability for real-time pruning operations, providing a balanced performance measure.

4.1.2. Annotation of Map

The deployment zone comprises a plant area exceeding 40 m, which is divided into two sections. Within the mapped area (Figure 13a), green RViz markers were placed at locations where the YOLOv8 instance segmentation model detected boxwood. The results of the boxwood annotation are illustrated in Figure 13b. After the annotation, waypoints were determined such that the robot's base link remained 1 m from the wall. The total annotated area was divided into action segments, each initialized to be 1 m long. To ensure comprehensive coverage of the entire annotated area, 40 waypoints were generated according to these conditions. An action segment is initialized as a test area for evaluation.

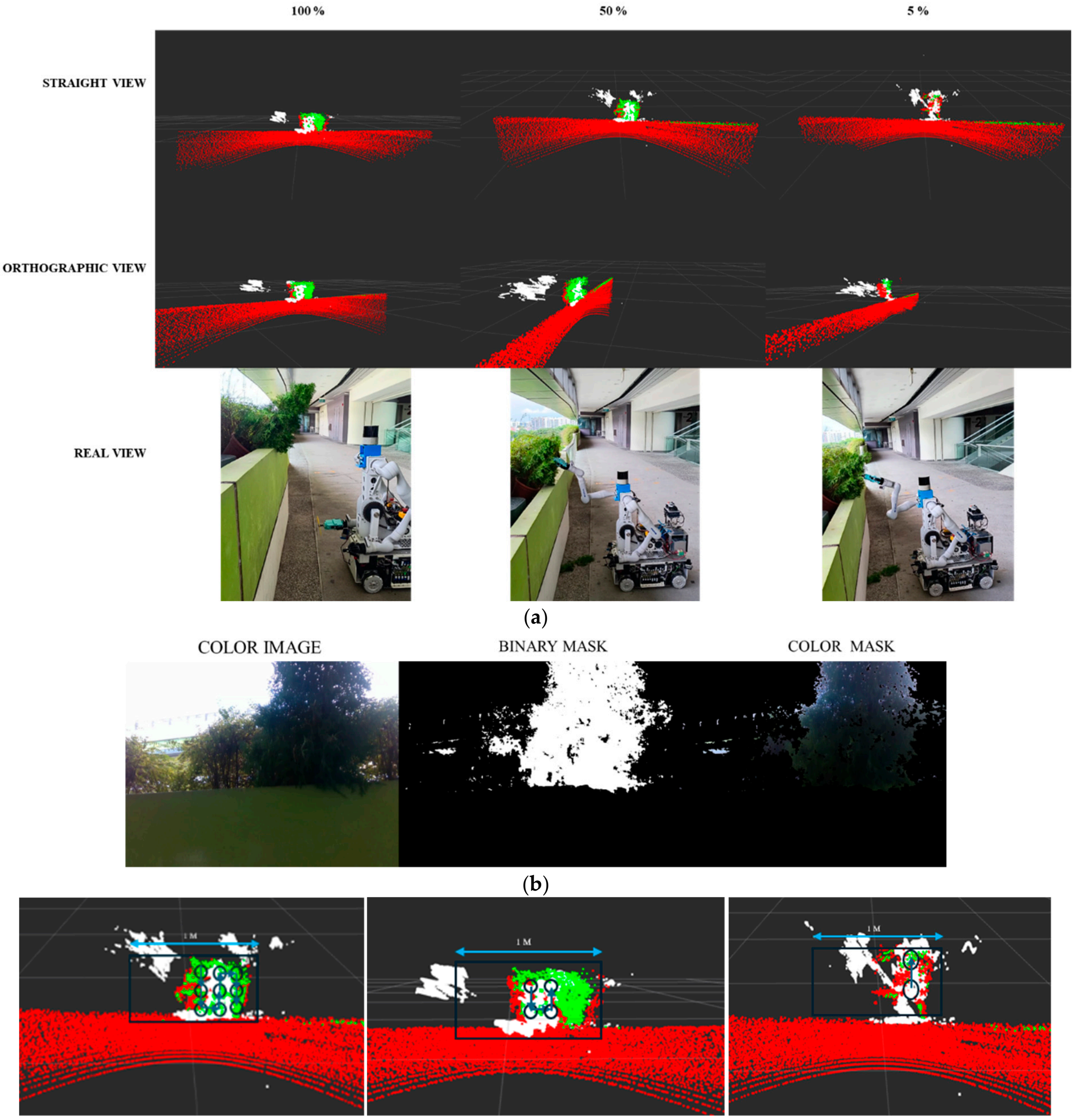


(**a**) (**b**)

Figure 13. (**a**) Deployment zone for testing the pruning algorithm; (**b**) detection of pruning subjects using YOLOv8.

4.1.3. Evaluation of Pruning Pattern

Figure 14a illustrates the cutting pattern followed in the experiment. Each point in the point cloud has a resolution of 1 mm. The red points highlight the wall plane, the green points represent the overflow points detected by the 3D LiDAR, and the white points indicate the plant points. The target points for pruning are the intersection of the green and white points, which are identified as the overflow plant points. The binary mask used for detecting plant points and the corresponding color mask are illustrated in Figure 14b. Table 2 presents the points in the 100%, 50%, and 5% scenarios. The number of waypoints assigned to the end effector gets reduced with each trimming session, as illustrated in Figure 14c. The number of positions to reach for trimming was 9, which gradually reduced to 2 when the scenario was nearly 5%. These positions are assigned from one end of the boundary to the other, with a regular interval of 20 cm, which is the blade size used. The end effector gets aligned to the center of the allocated point.



(a)

(b)

(c)

Figure 14. (**a**) Detection of target points at various stages of pruning operation; (**b**) original color image, binary mask, and color mask of the target zone; (**c**) waypoint estimation at various stages of pruning.

Table 2. Number of data points detected during the beginning (100%), middle (50%), and end of the pruning operation (5%).

Detection points	100%	50%	5%
Overflow points	8981	5451	1071
Plant points	13,456	8007	3246
Overflow plant points	7561	3780	207

4.2. Discussion

The experimental results of the KOALA robot showcase its accuracy and efficiency for automatic pruning, driven by the integration of advanced sensing systems such as LiDAR, the Intel RealSense depth camera, and the YOLOv8 object detection model. A demonstration video is attached to the Supplementary Materials depicting the working of the cross-functionality of these sensors, which play a crucial role in enhancing the robot's performance. Specifically, the RGB frame data, depth frame data from the RealSense camera, and 3D LiDAR data are merged to provide comprehensive environmental awareness, which allows for precise detection of the target boxwood plants. This sensor fusion ensures that the robot detects and accurately measures plant protrusions from the wall, helping to create a detailed 3D representation of the pruning area. The depth data from the RealSense camera aids in accurately gauging the distance between the pruning tool and the plant, while the RGB camera provides clear visual cues for object detection. This data is then combined with the 3D LiDAR's spatial scanning, which helps define the pruning points by assessing the plant's dimensions and density. This combined data enables the system to navigate complex environments with higher precision, significantly reducing noise and uncertainty that could arise if only one type of sensor were used.

The YOLOv8 model trained with the boxwood plant dataset achieved a mAP of 0.985 at a 50% IoU threshold, indicating its high accuracy in detecting target plants. The consistent performance of the model in the range of the IoU thresholds enabled the reliable classification of plants for pruning. Identifying and defining target areas on the map, as seen in Figure 13, demonstrates the system's ability to accurately locate plants in the environment.

Tests with known overflow points of 100%, 50%, and 5% demonstrate the system's flexibility to meet pruning requirements. As seen in Table 2, the number of detected points decreases as the cutting operation proceeds, reflecting the ability of the robot to cut overgrowth areas. The number of waypoints assigned to the end effector decreased from 100% at the beginning to 5% at the end. This reduction in waypoints indicates that reducing the cutting work during movement systematically improves the robot's operational efficiency because less movement is required.

The pruning sample, illustrated in Figure 14, demonstrates the system's accuracy in identifying the intersection among overflow points, which serve as the goal areas for pruning. The robot's capacity to dynamically modify the number of waypoints as the pruning operation progresses helps optimize strength consumption and time performance, ensuring that most superficial necessary regions are pruned.

Although the proposed system can identify targets and precisely prune them, operating robotic arms in urban landscapes is usually accompanied by various performance limitations. Several limitations were encountered in the evaluation phase of the precision pruning robot, and these restrictions repeatedly caused the robot to stop working. These restrictions include the following:

- The existence of plant stems with very large and unpredictable diameters in hedges.
- The existence of dried and completely woody stems of plants inside the hedge.
- The existence of shrub stems planted inside the hedge like flowers.
- The irregularity of the surface makes it difficult to move the robot, causing slippage.

Future research directions include the following:

- Synchronizing the workflow of the arms for maximum area coverage.

- Identifying the lowest-cost path for the given data point density, as well as investigations of noise handling and outlier removal techniques.
- Conducting a comprehensive cost-benefit analysis compared to manual pruning.
- Performing rigorous testing in outdoor environments to investigate ways to enable the robot to work alongside human workers safely and effectively, leveraging collaborative robotics principles.
- Implementing machine learning models that predict wear and tear on the robot's components, leading to proactive maintenance and reducing downtime.
- Developing compensation techniques for combatting embedded drift in performance in dynamic environments.

5. Conclusions

This paper introduces a dual-arm precision pruning robot with a cross-functionality sensor fusion approach for automated landscaping tasks. The robot combines LiDAR data for safe navigation, detecting paths and walls in the environment, with an RGB-D camera powered by the YOLOv8 object detection algorithm to precisely identify and target boxwood plants that require pruning. Fusing data from these sensors allows the robot to assess the density of overgrown areas and plan optimized pruning paths with high precision. The robot operates at pruning speeds of 10–50 cm/s and a maximum travel speed of 0.5 m/s, enabling it to cover large areas efficiently. It is equipped with a robust base capable of lifting 400 kg, ensuring stability and versatility for various applications. Furthermore, the robot can work continuously for up to 4 h, making it a reliable solution for extended operations in various environments.

The main contributions of this work include the development of a precision pruning algorithm with a pruning detection rate of 98.5% and an accuracy of 95%, achieved through the integration of a camera, depth sensors, and LiDAR data. This system significantly enhances operational efficiency by automating repetitive tasks, reducing worker injury risk, and improving pruning operation precision through advanced object detection. Moreover, the robot's scalable design enables its adaptation to various pruning tasks, demonstrating its potential in landscape maintenance by reducing labor requirements and increasing the quality and accuracy of the work performed. These contributions highlight the KOALA robot's value as an innovative tool for automating pruning tasks in landscaping maintenance.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/agriculture14101852/s1.

Author Contributions: Conceptualization, C.V. and P.K.C.; methodology, C.V. and P.K.C.; software, S.J. and C.V.; Writing, S.J., C.V. and P.K.C.; resources, S.P., M.R.E. and W.Y.; supervision, M.R.E.; project administration, W.Y. and M.R.E.; funding acquisition, M.R.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Robotics Program, under its National Robotics Program (NRP) BAU, Ermine III: Deployable Reconfigurable Robots, Award No. M22NBK0054, and is also supported by an SUTD-ZJU Thematic Research Grant, Grant Award No. SUTD-ZJU (TR) 202205.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy concerns, proprietary or confidential information, and intellectual property belonging to the organization that restricts its public dissemination.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sezavar, N.; Pazhouhanfar, M.; Van Dongen, R.P.; Grahn, P. The importance of designing the spatial distribution and density of vegetation in urban parks for increased experience of safety. *J. Clean. Prod.* **2023**, *403*, 136768. [CrossRef]
2. Montobbio, F.; Staccioli, J.; Virgillito, M.E.; Vivarelli, M. Labour-saving automation: A direct measure of occupational exposure. *World Econ.* **2024**, *47*, 332–361. [CrossRef]
3. Cheng, C.; Fu, J.; Su, H.; Ren, L. Recent advancements in agriculture robots: Benefits and challenges. *Machines* **2023**, *11*, 48. [CrossRef]
4. Borrenpohl, D.; Karkee, M. Automated pruning decisions in dormant sweet cherry canopies using instance segmentation. *Comput. Electron. Agric.* **2023**, *207*, 107716. [CrossRef]
5. Dong, X.; Kim, W.-Y.; Yu, Z.; Oh, J.-Y.; Ehsani, R.; Lee, K.-H. Improved voxel-based volume estimation and pruning severity mapping of apple trees during the pruning period. *Comput. Electron. Agric.* **2024**, *219*, 108834. [CrossRef]
6. Wang, Z.; Yang, C.; Che, R.; Li, H.; Chen, Y.; Chen, L.; Yuan, W.; Yang, F.; Tian, J.; Wang, B. Assisted Tea Leaf Picking: The Design and Simulation of a 6-DOF Stewart Parallel Lifting Platform. *Agronomy* **2024**, *14*, 844. [CrossRef]
7. Jun, J.; Kim, J.; Seol, J.; Kim, J.; Son, H.I. Towards an efficient tomato harvesting robot: 3D perception, manipulation, and end-effector. *IEEE Access* **2021**, *9*, 17631–17640. [CrossRef]
8. Arad, B.; Balendonck, J.; Barth, R.; Ben-Shahar, O.; Edan, Y.; Hellström, T.; Hemming, J.; Kurtser, P.; Ringdahl, O.; Tielen, T. Development of a sweet pepper harvesting robot. *J. Field Robot.* **2020**, *37*, 1027–1039. [CrossRef]
9. Ling, X.; Zhao, Y.; Gong, L.; Liu, C.; Wang, T. Dual-arm cooperation and implementing for robotic harvesting tomato using binocular vision. *Robot. Auton. Syst.* **2019**, *114*, 134–143. [CrossRef]
10. SepúLveda, D.; Fernández, R.; Navas, E.; Armada, M.; González-De-Santos, P. Robotic aubergine harvesting using dual-arm manipulation. *IEEE Access* **2020**, *8*, 121889–121904. [CrossRef]
11. Li, T.; Xie, F.; Zhao, Z.; Zhao, H.; Guo, X.; Feng, Q. A multi-arm robot system for efficient apple harvesting: Perception, task plan and control. *Comput. Electron. Agric.* **2023**, *211*, 107979. [CrossRef]
12. Xiong, Y.; Ge, Y.; Grimstad, L.; From, P.J. An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation. *J. Field Robot.* **2020**, *37*, 202–224. [CrossRef]
13. Mu, L.; Cui, G.; Liu, Y.; Cui, Y.; Fu, L.; Gejima, Y. Design and simulation of an integrated end-effector for picking kiwifruit by robot. *Inf. Process. Agric.* **2020**, *7*, 58–71. [CrossRef]
14. Li, K.; Huo, Y.; Liu, Y.; Shi, Y.; He, Z.; Cui, Y. Design of a lightweight robotic arm for kiwifruit pollination. *Comput. Electron. Agric.* **2022**, *198*, 107114. [CrossRef]
15. Silwal, A.; Davidson, J.R.; Karkee, M.; Mo, C.; Zhang, Q.; Lewis, K. Design, integration, and field evaluation of a robotic apple harvester. *J. Field Robot.* **2017**, *34*, 1140–1159. [CrossRef]
16. De-An, Z.; Jidong, L.; Wei, J.; Ying, Z.; Yu, C. Design and control of an apple harvesting robot. *Biosyst. Eng.* **2011**, *110*, 112–122. [CrossRef]
17. Zhang, Z.; Kayacan, E.; Thompson, B.; Chowdhary, G. High precision control and deep learning-based corn stand counting algorithms for agricultural robot. *Auton. Robot.* **2020**, *44*, 1289–1302. [CrossRef]
18. Rovira-Más, F.; Saiz-Rubio, V.; Cuenca-Cuenca, A. Augmented perception for agricultural robots navigation. *IEEE Sens. J.* **2020**, *21*, 11712–11727. [CrossRef]
19. Kim, K.; Deb, A.; Cappelleri, D.J. P-AgBot: In-Row & Under-Canopy Agricultural Robot for Monitoring and Physical Sampling. *IEEE Robot. Autom. Lett.* **2022**, *7*, 7942–7949. [CrossRef]
20. Devanna, R.; Matranga, G.; Biddoccu, M.; Reina, G.; Milella, A. Automated plant-scale monitoring by a farmer robot using a consumer-grade RGB-D camera. In Proceedings of the Multimodal Sensing and Artificial Intelligence: Technologies and Applications III, Munich, Germany, 26–29 June 2023; pp. 299–305. Available online: https://iris.cnr.it/handle/20.500.14243/439650 (accessed on 10 September 2024).
21. Song, P.; Li, Z.; Yang, M.; Shao, Y.; Pu, Z.; Yang, W.; Zhai, R. Dynamic detection of three-dimensional crop phenotypes based on a consumer-grade RGB-D camera. *Front. Plant Sci.* **2023**, *14*, 1097725. [CrossRef]
22. Jayasuriya, N.; Guo, Y.; Hu, W.; Ghannoum, O. Machine vision based plant height estimation for protected crop facilities. *Comput. Electron. Agric.* **2024**, *218*, 108669. [CrossRef]
23. Kamandar, M.R.; Massah, J.; Jamzad, M. Design and evaluation of hedge trimmer robot. *Comput. Electron. Agric.* **2022**, *199*, 107065. [CrossRef]
24. Kaljaca, D.; Mayer, N.; Vroegindeweij, B.; Mencarelli, A.; Van Henten, E.; Brox, T. Automated boxwood topiary trimming with a robotic arm and integrated stereo vision. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 5542–5549. [CrossRef]
25. van Marrewijk, B.M.; Vroegindeweij, B.A.; Gené-Mola, J.; Mencarelli, A.; Hemming, J.; Mayer, N.; Wenger, M.; Kootstra, G. Evaluation of a boxwood topiary trimming robot. *Biosyst. Eng.* **2022**, *214*, 11–27. [CrossRef]
26. He, F.; Fu, C.; Shao, H.; Teng, J. An image segmentation algorithm based on double-layer pulse-coupled neural network model for kiwifruit detection. *Comput. Electr. Eng.* **2019**, *79*, 106466. [CrossRef]
27. Sa, I.; Ge, Z.; Dayoub, F.; Upcroft, B.; Perez, T.; McCool, C. Deepfruits: A fruit detection system using deep neural networks. *Sensors* **2016**, *16*, 1222. [CrossRef]

28. Ji, W.; Gao, X.; Xu, B.; Chen, G.; Zhao, D. Target recognition method of green pepper harvesting robot based on manifold ranking. *Comput. Electron. Agric.* **2020**, *177*, 105663. [CrossRef]
29. de Luna, R.G.; Dadios, E.P.; Bandala, A.A.; Vicerra, R.R.P. Size classification of tomato fruit using thresholding, machine learning, and deep learning techniques. *AGRIVITA J. Agric. Sci.* **2019**, *41*, 586–596. [CrossRef]
30. Williams, H.A.; Jones, M.H.; Nejati, M.; Seabright, M.J.; Bell, J.; Penhall, N.D.; Barnett, J.J.; Duke, M.D.; Scarfe, A.J.; Ahn, H.S. Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms. *Biosyst. Eng.* **2019**, *181*, 140–156. [CrossRef]
31. Rahul, M.; Rajesh, M. Image processing based automatic plant disease detection and stem cutting robot. In Proceedings of the 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 20–22 August 2020; pp. 889–894. [CrossRef]
32. Li, C.; Zhang, H.; Wang, Q.; Chen, Z. Influencing factors of cutting force for apple tree branch pruning. *Agriculture* **2022**, *12*, 312. [CrossRef]
33. Varghese, R.; Sambath, M. YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness. In Proceedings of the 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), Chennai, India, 18–19 April 2024; pp. 1–6. [CrossRef]
34. Kaljaca, D.; Vroegindeweij, B.; Van Henten, E. Coverage trajectory planning for a bush trimming robot arm. *J. Field Robot.* **2020**, *37*, 283–308. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.