DEVELOPMENT AND TESTING OF AUTOMATED CONTROL SYSTEM FOR SEA BUCKTHORN BERRY HARVESTING ROBOT "AGROBOT"

Andrejs Stupans, Pavels Maksimkins, Armands Senfelds, Leonids Ribickis
Riga Technical University, Latvia
andrejs.stupans@rtu.lv, pavels.maksimkins@rtu.lv, armands.senfelds@rtu.lv, leonids.ribickis@rtu.lv

Abstract. The paper describes the control system of an autonomous agriculture robot and evaluates its operation using a laboratory stand and a digital environment. AgroBot is an autonomous sea buckthorn berry harvesting robot. It automatically finds the bush branches to harvest, cuts them, and stores them in a box. AgroBot consists of a 3 DOF (Degrees of Freedom) Cartesian mobile platform and a Hyundai HH7 industrial robot arm with 6 DOF. The control system is specifically designed for real-time operation, enabling AgroBot to adapt to dynamic environmental conditions (wind, varying light) that complicate target tracking. The control system consists of two modules operating as two separate programs. The first module is the Computer Vision Module (CVM) which has high-level control of AgroBot operation. It uses feed from cameras to find the cutting spot and sends commands to approach, cut, store, and search. The module is written in Python. The second module is the Robot Control Module (RCM) which receives high-level commands from CVM and manages lower-level control of the Hyundai Controller (HC). RCM calculates the robot trajectory to a target position, communicates with both CVM and HC in parallel threads and handles errors. As HC requires time-critical control, RCM is written in C + + . RCM and CVM are running on one computer communicating via sockets. As the mobile platform hardware is in the development stage and is not available at the moment, the digital twin of the robot is created to test the system's performance in a simulated environment. The input to the digital twin is the same as for the actual robot. It is the x, y, z position and orientation A, B, C using Euler angles. The digital twin visualization is developed in the Unity game engine. Matlab Robotic Toolbox is used with the Levenberg-Marquardt solver algorithm to calculate inverse kinematics of 9 DOF robot. The paper focuses on the Robot Control Module architecture and control system's testing.

Keywords: manipulators, smart agriculture, agricultural robots.

Introduction

Rapid developments in the field of robotics brought modern technologies to the agricultural sector. Historically, agricultural robots started with simple automated systems designed to perform very specific tasks. Over time the technology evolved into more complex multipurpose systems capable of navigating fields [1; 2], decision-making [3], and executing tasks with precision previously unattainable by human labour alone [4]. Serial manipulator path planners can take up to several seconds to calculate the path to the target. In such a dynamic environment, where target acquisition is disturbed by nature-induced factors (sun rays, wind etc.) it is considered too slow to react and adapt to the disturbances. The control algorithm of AgroBot is developed to adapt to the rapidly changing environment.

Traditional farming practices cannot reach maximum productivity. One of the main reasons is the lack of human labour to cope with production at the peak of the harvest season [5]. Sea buckthorn is hard to harvest because berries are very tightly bunched along the branches. Big thorns also make manual harvesting unattractive for workers. That is why it is important to automate the process. Automated harvesting can significantly contribute to more sustainable farming. Recent developments on current agricultural challenges are well described in [6], where the need for improved locomotion systems, more effective computer vision algorithms and sensor system capabilities is emphasized to overcome the problems of agricultural environment. Moreover, the integration of mobile robot systems into a farming process [7] provides a more detailed review of the transformative impact of UAVs (Unmanned Aerial Vehicles) and UGVs (Unmanned Ground Vehicles) on agricultural processes, highlighting the increased crop monitoring precision and improved resource management.

Recent developments in the mechanical harvesting of sea buckthorn are reviewed in [8] proposing various methods that enhance the efficiency of the harvesting process and reduce human labour dependency, which provides a foundation for more advanced automated robotic solutions. Significant steps in sea buckthorn harvesting techniques have been documented in [9], providing a review of the approaches from traditional manual methods to more recent mechanical solutions, underlining the growing need for innovations in this field.

DOI: 10.22616/ERDev.2024.23.TF175

There are three main methods for mechanized harvesting of sea buckthorn: by vacuum, vibration and cutting [8]. For Latvian sea buckthorn farms, the most common practice is cutting. To harvest sea buckthorn, the branches with berries are cut and placed in a box. Then branches are frozen, and berries separate easily. Not all branches can be cut, there are different methods how to process a bush so it will be able to grow new berries next year. The decision process is handled by computer vision and is fully autonomous [10]. Cutting and storing are jobs for a manipulator control system, also fully autonomous. The only part that is not autonomous is moving AgroBot between the bushes. This work is carried out by an operator using a tractor to pull the AgroBot.

The paper is structured into seven sections. In the first section, an overview of the harvesting system is presented. The next section continues with the robot control algorithm, discussing tracking functionality, communication and coordinate frame transformation. The third section presents the harvesting algorithm. In the subsequent sections, a testing methodology is provided. Then, integration with the digital twin technology is discussed. Finally, the results of preliminary tests are presented. The paper concludes with future plans for improvements.

Overview of the harvesting system

AgroBot consists of a 3 DOF (Degrees of Freedom) Cartesian mobile platform and a Hyundai HH7 industrial robot arm with 6 DOF. AgroBot has 9 DOF in total. The system is powered by a 230V three-phase generator available on the tractor that pulls the platform. The AgroBot hardware setup is shown in Fig. 1.

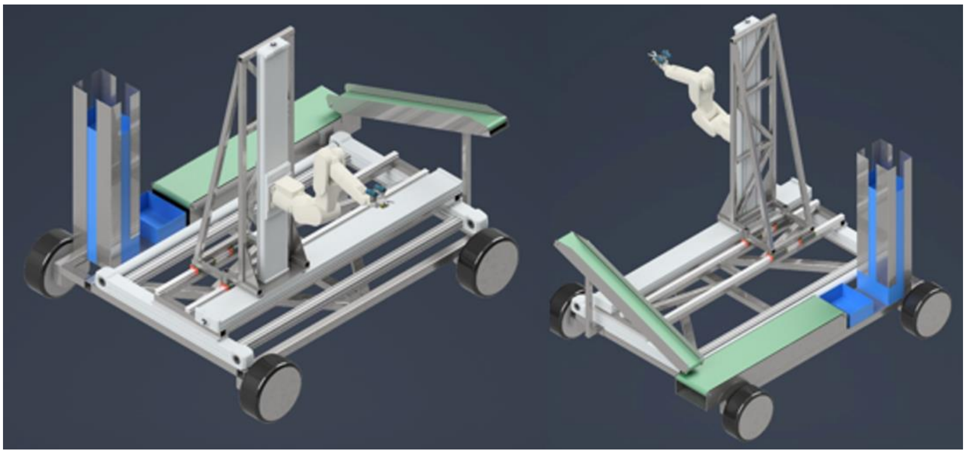


Fig. 1. **Mechanical system of AgroBot**

AgroBot uses eye-in-hand setup. The “eye” of the AgroBot system is the Intel RealSense D435i depth camera. It is used to record the depth map of the space and a corresponding RGB picture. These data then get processed on the Control PC. It runs the Computer Vision Module (CVM) and Robot Control Module (RCM). RCM, CVM and HC are communicating via sockets.

CVM is responsible for processing the image gathered by the Intel RealSense depth camera, deciding on where the branches and cutting points are located and providing this information to RCM. CVM provides the target position in the camera reference frame.

RCM recalculates the target coordinates in the world reference frame, calculates the robot trajectory from an actual position to a target position and handles the motion of the robot tool (scissors) towards the cutting point. RCM communicates with both CVM and HC in parallel threads. RCM uses an inverse kinematics solver provided by HC. It can control up to 8 axes: 6 for manipulator and 2 external. The 9th external axis will be controlled by RCM directly as shown in Fig. 2.

Robot motion is executed by the Hyundai Hi-5a Controller (HC). The decision to divide the control program into two modules is justified by the requirements of the Hyundai Controller (HC) and developer team experience. As HC requires time-critical control, RCM is written in C + + to ensure the reliability and speed of the system. CVM, however, runs on Python because the developer team has more experience with Python API for RealSense SDK.

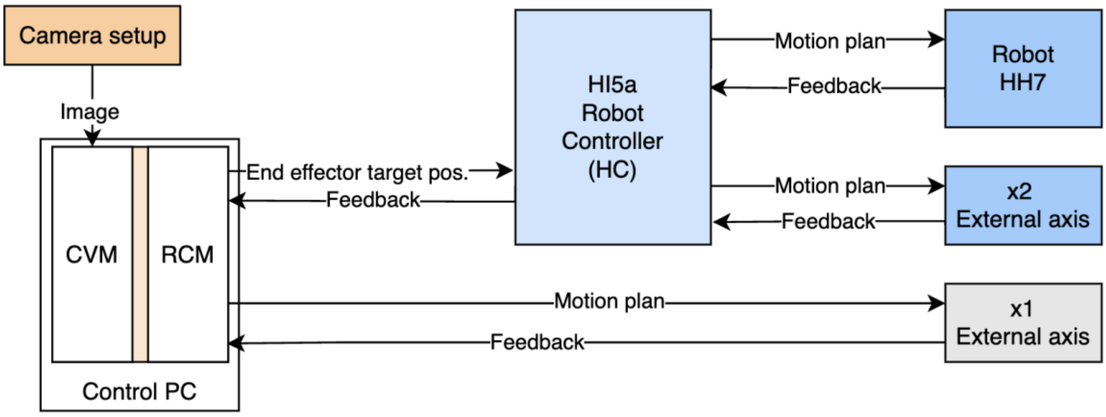


Fig. 2. **Overview of the AgroBot control system**

Robot control

The control of the Hyundai HH7 industrial robot arm is covered by the integrated HC functionality called “On-line Tracking” (OnLTrack). OnLTrack functionality allows the control of the robot through the UDP commands. HC serves as a UDP client, it receives the incremental position and orientation of the end-effector as an input, calculates the robot axes’ configuration using a built-in inverse kinematics solver and moves the axes to reach the target position. HC expects to receive a new position every 5 milliseconds and executes the motion within this time period. It is important to send data consistently to avoid shaking motion, which can lead to a bad camera image quality and instability of the platform.

RCM is running the UDP server. It gets the target position from CVM, calculates the trajectory, and sends increments to the HC client. The server also receives back the actual position of the robot provided by HC.

Communication between CVM and RCM is done via TCP/IP. The speed is sacrificed for the reliability that TCP provides because the amount of data transferred between the modules is rather small. If a data packet with the important command is lost it has to be resent until successfully delivered. TCP provides this functionality and UDP does not.

Three reference frames are defined for the robot positioning – camera, tool, and world shown in Fig.3. As RCM receives target coordinates in the camera reference frame, it should be recalculated for the world reference frame in which the robot operates. Recalculation of reference frames is done using the homogeneous transformation matrix H that is a product of multiplying H_1 and H_2, where H_2 is fixed and has a value shown in (1) and H_1 value is dynamically changing according to the robot state. The transformation from Euler to Rotation matrix is needed, because the Hyundai controller input/output is in Euler angles. The rotation matrix and translation are converted into the homogeneous transformation matrix H_1. The *”Eigen3”* C++ library is used for fast matrix operations.

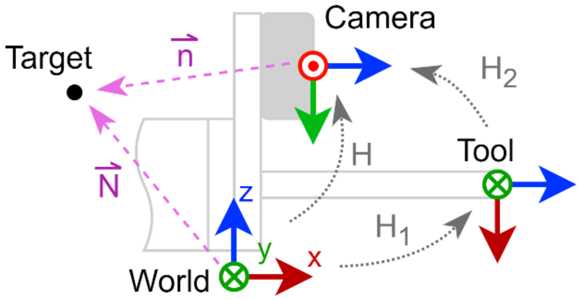


Fig. 3. **Reference frame definition for AgroBot (figure is not-to-scale)**

$$\vec{N} = H\vec{n} = H_1 H_2 \vec{n} = H_1 \begin{matrix} 0 & 1 & 0 & -0.04 \\ -1 & 0 & 0 & 0.033 \\ 0 & 0 & 1 & -0.29 \\ 0 & 0 & 0 & 1 \end{matrix} \vec{n}, \tag{1}$$

where $\vec{N}$ – target position in the world reference frame;
$\vec{n}$ – target position in the camera reference frame;
H – homogeneous transformation matrix from the world to camera reference frame;
H_1 – homogeneous transformation matrix from the world to tool reference frame;
H_2 – homogeneous transformation matrix from the tool to camera reference frame.

Considering that the platform will not be perfectly stable during the robot movements and will slightly change its inclination, a correction of the robot target coordinates in relation to the position of the platform should be done. In AgroBot it will be achieved with an inertial measurement unit (IMU) that will cancel out the platform's changed reference frame. To further reduce the effect of the instability of the platform, the robot will take a transporting position while the platform is moved to the next bush. That will lower the platform centre of mass making it more stable, while the robot arm will be located within the platform boundaries reducing the risk of damaging buckthorn bushes and the robot itself.

Algorithm

The algorithm for the strawberry harvesting robot, suggested in [11], first analyses all targets, decides on the processing order, and only then executes the harvesting process. This approach is not suitable for AgroBot as the sea buckthorn branches can move in the presence of wind and manipulator intervention, making it impossible to find and record all the cutting point coordinates beforehand. That is why AgroBot is cutting each branch sequentially adapting to the changes in real-time. At first, CVM finds all possible branches to cut and chooses the best option to proceed. Then the robot cuts the branch, stores it and continues with another branch until all branches fitting the criteria are cut. When the bush is fully processed, AgroBot is moved to the next one. Fig. 4 shows the harvesting cycle for one sea buckthorn bush.

Operation of Agrobot is divided into 4 main stages – *Search*, *Approach*, *Cut* and *Store*. RCM operates according to the active stage. All the high-level decisions about Argrobot operation are made in CVM. It decides which stage to activate and sends the commands to RCM. RCM then executes the low-level control of the operation. During the *Search* stage RCM moves the camera around, so CVM can find branches to be cut. When CVM finds the branch, it changes the active stage to *Approach*. RCM plans the robot trajectory and executes the motion to approach the cutting point of the branch. When the approach is complete, CVM changes the active stage to *Cut*. RCM executes the cut procedure and sends back completion confirmation. Then CVM changes the active stage to *Store*. RCM plans the robot trajectory, executes the motion to the storage box and puts the branch there. Then CVM repeats the cycle starting from the *Search* stage until the bush is fully processed. The paper focuses on the *Approach* stage execution.

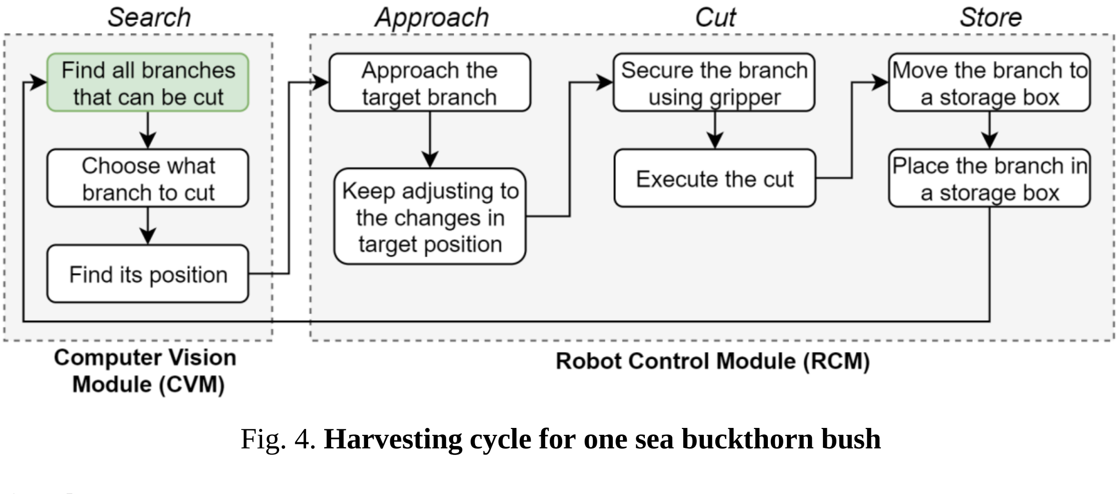


Fig. 4. **Harvesting cycle for one sea buckthorn bush**

Testing the system

Even though the mobile platform is still in the development stage, some preliminary tests in the laboratory and the digital environment were conducted. The manipulator was mounted on a static platform connected to HC powered from the grid through a transformer. Fig. 5 shows the test setup of the AgroBot system. For testing, the substitute of CVM was used, because the original computer vision model was trained on the sea buckthorn dataset and can only detect the berry branches which are not available during non-season. As the main focus of the study is RCM's target approaching algorithm, the alternative computer vision object detection system was used with functionality similar to CVM but detecting round object position as shown in Fig. 6.

The proposed real-time target approaching and following algorithm was tested in two modes. First, in stationary mode, a red marker was placed in the camera view and the robot had to reach the target position (defined by the centre of the marker) with the tool end-point. Second, in motion mode, the marker was constantly moved, testing the robot ability to follow the marker.

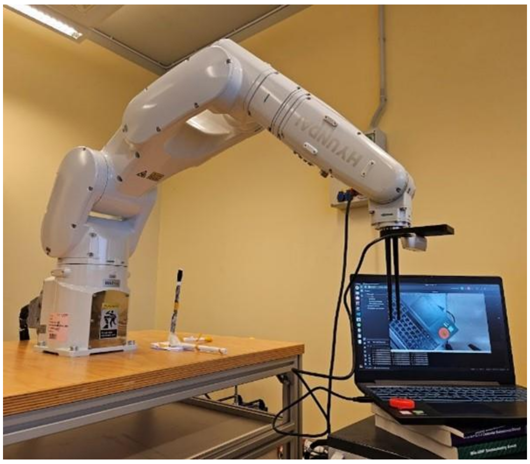


Fig. 5. **Initial testing stand**

3D: [0.0092, −0.0288, 0.2630]

Fig. 6. **Computer vision tracking the red target and calculating the 3D position**

As harvesting season lasts only for 3-4 weeks once a year, there is a need to be able to test the AgroBot between the seasons. When field tests are not an option, the use of computer modelling is a good substitute as done by developers of the melon-harvesting robot [12]. For that purpose, a digital environment for AgroBot is created using Matlab and Unity. A kinematical model of the Hyundai HH7 manipulator is created in Matlab to achieve a real-time inverse kinematics solution. The precise 3D model of the HH7 robot is created in Unity and is used for visualization purposes as shown in Fig. 8. Unity was chosen to replace Matlab renderer which is harder to navigate and has a lower framerate and graphics level.

Fig. 7 shows how Matlab, Unity and RCM are set up. RCM sends the manipulator target position to Matlab, which solves inverse kinematics in real-time using the Levenberg-Marquardt algorithm. The solution is the manipulator 6-axes configuration that is sent to Unity also via UDP. The script in Unity receives the configuration and moves the manipulator axes to corresponding positions. Interface from RCM to Matlab is developed in such a way that Matlab can be changed with HC without major modifications in RCM. Only IP and Port configuration should be changed. This allows to develop CVM and RCM software for a real HC, but test it using the digital twin of the robot. At the moment, a very simple environment is used for motion trajectory visualization to easier tune RCM without physical access to the robot. However, in the future, a full-scale AgroBot system with simulated camera input will be integrated into the digital environment.

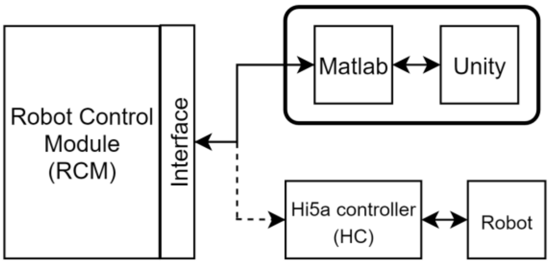
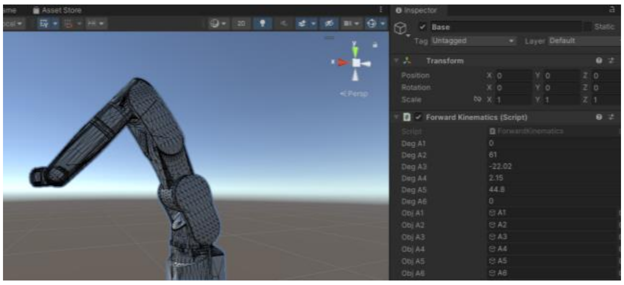


Fig. 7. **Digital testing environment setup**



Base
Tag Untagged
Layer Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Forward Kinematics (Script)
Script ForwardKinematics
Deg A1 0
Deg A2 61
Deg A3 -22.02
Deg A4 2.15
Deg A5 44.8
Deg A6 0
Obj A1 A1
Obj A2 A2
Obj A3 A3
Obj A4 A4
Obj A5 A5
Obj A6 A6

Fig. 8. **HH7 robot model in Unity environment**

Results and discussion

Preliminary tests show that control of the Hyundai HH7 robot using “On-Line Tracking” functionality is reliable even considering the high frequency of communication needed for the smooth operation of the robot arm. RCM ability to handle communication with CVM and HC shows the robustness of the system architecture.

As mentioned in the Intel RealSense D435i datasheet, the minimal focusing point of the camera stereo module is 28cm. However, during the testing, target tracking was working reliably only when the object was further than 30cm from the sensor of the camera. It is not considered a significant drawback since the scissor dimensions are planned to be almost out of the blind zone of the camera. However, additional insights into the problem should be gained. The observed robot positioning error of the tracked target was < 1cm. Shadowing, artificial lighting and focal distortion of the camera sensor could be a possible cause of the tracking error. It is expected that the error of tracking will increase during the field tests in the agricultural environment due to the changing weather conditions (wind, lighting etc.) and instability of the platform that carries the robot arm.

When a stationary target position is reached, the manipulator end-effector remains still, indicating that the target tracking is working efficiently. When the marker is moving, changing the target position, the manipulator successfully follows along. The maximum speed of the robot during the motion test was found to be 50 mm/s. When the speed was increased even further, the manipulator motion near the target became more chaotic, undesirable vibrations appeared, and the camera image jitter increased, leading to unstable behaviour and loss of the target tracker. The causes of such behaviour are to be discovered in future studies.

Future plans

The project is in the development phase. Future plans for improvement include mounting additional cameras to the platform (incorporating an inertia measurement unit inside them for sensing the orientation of the platform and correcting the end-effector position in the world coordinate frame. Additional cameras are to be integrated into the computer vision solution giving images from different angles, which will increase the precision and reliability of the harvesting system. Gathered image data will be used to train the CVM decision-making algorithm. Additional measures include evaluating the correctness of the platform wheel size to provide off-road driving capabilities and stability of the platform. During the field tests it is also planned to verify the efficiency of the designed scissors. HH7 industrial arm singularity points should be evaluated to provide more robust motion control algorithms. Ongoing modifications will further increase the efficiency of the sea buckthorn harvesting robot.

Conclusions

The research findings underscore the robustness of the proposed real-time control system, particularly in the interaction between the Robot Control Module and Computer Vision Module ensuring a reliable control loop within a 5ms timeframe. AgroBot ability to approach a target position was evaluated during laboratory tests with both a stationary target and a target in motion. Results indicate that the system is capable of target following in a dynamically changing environment, which is essential for sea buckthorn harvesting tasks. Additionally, the virtual environment was developed for robot motion visualization helping tune the Robot Control Module without physical access to the robot.

However, some challenges have been identified such as the camera stereo module focusing limitations or robot vibrations at higher operation speeds. These areas have to be investigated further. Addressing these challenges is crucial to enhancing the system performance and reliability during the field tests. Overall, the research provides valuable insights into the performance of the AgroBot control system, laying a foundation for further development towards practical agricultural applications.

Acknowledgement

Publication is created with support of the Latvian Rural Development Program “Cooperation”, call 16.12 project Nr. 22-00-A01612-000021 “AgroBot harvester for cutting branches of fruit bushes”.

Author contributions

A. Stupāns, P. Maksimkins – methodology, software, testing, writing – draft preparation, A. Šenfelds – methodology, project administration, L.Ribickis – project administration, writing – review and editing. All authors have read and agreed to the published version of the manuscript.

References

[1] Bayar G., Bergerman M., Koku A.B. Konukseven E.I, “Localization and Control of an Autonomous Orchard Vehicle,” Computers and Electronics in Agriculture 115, 2015, p. 118.
[2] Bell J., MacDonald B. A., Ahn H. S., Scarfe A. J. “An Analysis of Automated Guided Vehicle Standards to Inform the Development of Mobile Orchard Robots’,” IFAC-PapersOnLine, 2016, pp. 475-480.
[3] Mehta S. S., Burks T. F. “Vision-based control of robotic manipulator for citrus harvesting,” Computers and Electronics in Agriculture, 2014, pp. 64-69.
[4] Visentin F., Cremasco S., Sozzi M., Signorini L., Signorini M., Marinello F.. Muradore R. “A Mixed-Autonomous Robotic Platform for Intra-Row and Inter-Row Weed Removal for Precision Agriculture,” Computers and Electronics in Agriculture, 2023.
[5] Rahmadian R., Widyartono M. “Autonomous Robotic in Agriculture: A Review,” 2020 Third International Conference on Vocational Education and Electrical Engineering (ICVEE), 2020, pp. 1-6.
[6] Oliveira L., Moreira A., Silva M. “Advances in Agriculture Robotics: A State-of-the-Art Review and Challenges Ahead,” Robotics 2021.
[7] Yépez-Ponce D., Salcedo J., Rosero-Montalvo P., Sanchis J. “Mobile robotics in smart farming: current trends and applications,” Frontiers in Artificial Intelligence, 2023.
[8] Fu L., Su H., Li R. Cui Y. “Harvesting technologies for sea buckthorn fruit,” Engineering in Agriculture Environment and Food, 2014.
[9] Mei D., Ma X., Fu F., Cao F. “Research Status and Development Prospects of Sea buckthorn (Hippophae rhamnoides L.) Resources in China,” Forests 2023.
[10] Timofejevs J., Potapovs A., Gorobetz M. “Computer Vision System for Autonomous Sea Buckthorn Harvesting Robot,” 2023 IEEE 64th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), 2023, pp. 1-5.
[11] Hayashi S., Shigematsu K., Yamamoto S., Kobayashi K., Kohno Y., Kamata J., Kurita M. “Evaluation of a Strawberry-Harvesting Robot in a Field Test,” Biosystems Engineering, 2010, p. 160.
[12] Zion B., Mann M., Levin D., Shilo A., Rubinstein D., Shmulevich I. “Harvest-Order Planning for a Multiarm Robotic Harvester,” Computers and Electronics in Agriculture, 2014, pp. 75-81,