

Robust Image-based Visual Servoing for Autonomous Row Crop Following with Wheeled Mobile Robots*

Gustavo B. P. Barbosa¹, Eduardo C. da Silva¹, Antonio C. Leite²

Abstract—In this work, we present a new robust vision-based controller for wheeled mobile robots, equipped with a fixed monocular camera, to perform autonomous navigation in agricultural fields accurately. Here, we consider the existence of uncertainties in the parameters of the robot-camera system and external disturbances caused by high driving velocities, sparse plants, and terrain unevenness. Then, we design a robust image-based visual servoing (rIBVS) approach based on the sliding mode control (SMC) method for robot motion stabilization, even under the presence of such inaccuracies and perturbations. The vision-based controller, based on column and row primitives, is slightly modified to include a robustness term into the original feedback control laws to ensure successful row crop reaching and following tasks. We employ the Lyapunov stability theory to verify the stability and robustness properties of the overall closed-loop system. 3D computer simulations are carried out in the ROS-Gazebo platform, an open-source robotics simulator, using a differential-drive mobile robot (DDMR) in an ad-hoc developed row crop environment to illustrate the effectiveness and feasibility of the proposed control methodology.

I. INTRODUCTION

The growth of agricultural productivity has been driven since the last century, making farms larger and more complex. Consequently, agriculture has demanded technological innovation, such as intelligent machinery with decision-making capabilities, genetically modified seeds, and highly accurate sensors for monitoring the cultivation fields [1]. Autonomous navigation systems and self-driving vehicles have also emerged as an essential branch of precision agriculture and are becoming a highly relevant research area [2]. These cutting-edge technologies can help the farmers to reduce environmental impacts, minimize the use of pesticides, as well as lower production and logistics costs [3].

Several wheeled mobile robots (WMRs), equipped with sensor packages and different tools, have been developed to carry out various agricultural tasks. For example, spraying robots such as the ecoRobotix can be autonomously guided to follow crop rows using machine vision, GPS RTK and other sensors. Then, fast Delta robot arms move micro-spray nozzles attached to their end-effectors to apply herbicides on the detected weed plants selectively [4]. Monitoring robots, such

as the SoyBot, can navigate autonomously along crop rows in soybeans and cotton farms for plant health monitoring, mainly carrying out diseases and insect pests inventory [5]. Autonomous modular robots such as the Thorvald II can be re-configurable for most agricultural environments and carry out fruit harvesting in polytunnels, ultraviolet light treatment in greenhouses, and plant phenotyping in open fields [6].

Following this trend, several vision-based autonomous navigation techniques for vehicle guidance have been developed using monocular, stereo and RGB-D cameras [7]. An image-based visual servoing (IBVS) approach for robot navigation in urban environments was introduced in [8], where the authors design two primitive vision-based controllers which require less computing resources: a column controller for path reaching; and a row controller for path following. This solution, however, was applied for mobile robot control only in partially structured scenarios instead of open agricultural fields, which are commonly subject to variability in soil properties and crop plant density. In [9], the aforementioned IBVS approach was firstly employed for row crop following and two monocular cameras, mounted back and forth, were used to avoid the need for maneuvers in the headlands. However, as will be mentioned in our work, the IBVS column and row controllers introduce an unwanted term into the image error equation, which may lead to system instability when the desired driving velocity is increased.

In this context, a variety of robust controllers have been developed to reject perturbations and deal with unmodeled dynamics in visual servoing systems. Becerra *et al.* [10] have designed a sliding mode control (SMC) method to stabilize the pose of mobile robots equipped with an uncalibrated epipolar camera and to deal with singular configurations of the image Jacobian matrix, allowing for the vehicle to move directly toward the target. Due to their discontinuous nature, robust control laws can lead to an unwanted chattering effect caused by imperfect control switchings, which may excite high-frequency unmodeled dynamics [11]. Then, chattering attenuation can be reached by using continuous approximation for switching control laws, finite-time control (FTC) techniques [12] and second-order SMC approaches such as the super-twisting algorithm (STA) [13]. Becerra *et al.* [14] have introduced an STA-based visual servoing system for mobile robots, which uses uncalibrated camera models to successfully perform tracking tasks of online-generated image trajectories with unknown first-order derivative.

In this work, a new robust image-based visual servoing (rIBVS) approach for WMRs, equipped with a single monocular camera, is developed to increase the accuracy of

*This work was partially supported by the Brazilian National Council for Scientific and Technological Development (CNPq Brazil) and the Research Council of Norway through the project SFI Digital Food Quality (DigiFoods), Project number 309259.

¹Gustavo B. P. Barbosa and Eduardo C. da Silva are with the Department of Electrical Engineering, Pontifical Catholic University of Rio de Janeiro, Marquês de São Vicente 225, 22451-900 Rio de Janeiro RJ, Brazil. E-mail: gustavo.bertagna@ieee.org, edusilva@ele.puc-rio.br

²Antonio C. Leite is with the Faculty of Science and Technology, Norwegian University of Life Sciences, Universitetstunet 3, 1433 Ås, Norway. E-mail: acleite@ieee.org

the autonomous navigation task in agricultural fields. The rIBVS approach seeks to deal with model inaccuracies and disturbances due to the weak camera calibration or trajectory perturbations that appear at high driving velocities. Thereby, an SMC approach is used to include a robustness term into the classic IBVS approach based on two primitive controllers (column and row) in the image space [8]. We inserted the 3D CAD model of the SoyBot robot in a row crop agricultural field developed in the ROS-Gazebo platform, an open-source robotics simulator, for evaluating the image-based autonomous navigation solution in different scenarios. Simulation results are included to illustrate the performance and feasibility of the proposed robust vision-based controller.

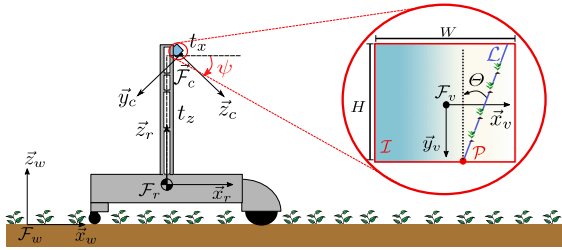


Fig. 1. The SoyBot robot and its coordinate frames: \mathcal{F}_v is the image frame and $s = (x_v, y_v, \theta_v) \in \mathbb{R}^3$ is the image features vector. Side view of the robot: \mathcal{F}_r is the robot frame, \mathcal{F}_w is the world frame, \mathcal{F}_c is the camera frame with an offset tilt angle ψ .

II. PROBLEM STATEMENT

Here, we consider the problem of row crop following by using a fixed monocular camera attached to the top of a wheeled mobile robot (WMR). We assume that the task of interest consists of autonomously guiding the robot over a single crop row (straight or curved), keeping its wheels in the midpoint between two adjacent rows. The SoyBot robot [5] is used for autonomous visual navigation in the presence of uncertainties and disturbances in the camera-robot system's parameters. During the row crop following, RGB images are continuously captured by the vision sensor. An image segmentation method, based on the combination of RANSAC and Hough transform algorithms [15], is then applied to find the best-fit line \mathcal{L} to represent the crop row and extract line features. Then, the following 2D image features set $\mathcal{L} = \{\mathcal{P}, \Theta\}$ is computed in the image frame \mathcal{F}_v : (i) $\theta_v \in (-\pi, \pi]$ is the coordinate of image feature angle Θ , denoting the angle between the line and the negative y_v -axis; (ii) $(x_v, y_v) \in \mathbb{R}^2$ are the coordinates of the image feature point \mathcal{P} , denoting the bottom (top) pixel row or left (right) pixel column. Therefore, the image features set can be denoted by the image features vector $s = (x_v, y_v, \theta_v) \in \mathbb{R}^{3 \times 1}$.

Afterward, a robust IBVS approach is designed to enable the robot to visually reach and follow a path on the ground defined by the crop rows [9]. An uncalibrated camera is mounted with an offset tilt angle $\psi \in (-\frac{\pi}{2}, \frac{\pi}{2}]$ with respect to the x -axis of the camera frame \mathcal{F}_c . In addition, the camera's optical center is positioned at the coordinates $(t_x, 0, t_z) \in \mathbb{R}^3$ with respect to the robot frame \mathcal{F}_r , as shown in Fig. 1. For designing the rIBVS controller for real-world agricultural applications, the following assumptions are considered: (A1)

the robot is located outside the crop row, so that both column and row primitives may be included into the control design; (A2) the image processing algorithm uses a region of interest (RoI) to find the best-fit line denoting the crop row to be monitored; (A3) the robot can travel with constant high driving velocities inside the crop field; and (A4) the terrain is rough as well as has plants sparsely distributed and irregularly located. Then, a robustness term will be included in the classic IBVS controller, based on column and row primitives [8], to cope with the external perturbations caused by assumptions (A3) and (A4). The new rIBVS controller will be split into twofold: a robust IBVS column controller (rIBVS-CC), to reach out the crop rows and maneuver the robot; (ii) a robust IBVS row controller (rIBVS-RC), to keep the robot following the path after reaching it.

III. MODELING OF THE ROBOT-CAMERA SYSTEM

Here, we briefly introduce the kinematic modeling of the differential-drive mobile robot (DDMR) and the image-based visual servoing (IBVS) system.

A. Kinematic Robot Model

The unicycle robot is a rigid body with a single orientable wheel and its configuration is described by the generalized coordinates $q = (x, y, \theta) \in \mathbb{R}^{3 \times 1}$, wherein $(x, y) \in \mathbb{R}^{2 \times 1}$ is the Cartesian coordinates of a given point of the wheel and $\theta \in \mathbb{R}$ is the wheel orientation with respect to the x -axis of the world frame \mathcal{F}_w . WMRs are subject to kinematic constraints, integrable or not (nonholonomic constraints), which reduce the mobility of the mechanical structure. For instance, the motion of a unicycle robot is characterized by a pure rolling constraint given by $\dot{x} \sin \theta - \dot{y} \cos \theta = 0$, which means that the point velocity $(\dot{x}, \dot{y}) \in \mathbb{R}^{2 \times 1}$ is zero in the direction orthogonal to the sagittal axis of the wheel [16]. In this context, the kinematic model of the unicycle robot is expressed in terms of the null space of the kinematic constraint as:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega, \quad (1)$$

where the robot inputs $v \in \mathbb{R}$ and $\omega \in \mathbb{R}$ are the driving and steering velocities respectively. However, a single wheeled robot has serious problems of balance in static conditions. That is why there are vehicles kinematically equivalent to a unicycle robot which use two or more wheels to provide higher mechanical stability, such as the DDMR. Its configuration is described by the same generalized coordinates of the unicycle robot, where $(x, y) \in \mathbb{R}^{2 \times 1}$ is the Cartesian coordinates of the midpoint between the two wheel centers and θ is the common orientation of the fixed wheels (Fig. 2).

Hence, the kinematic model in Eq. (1) is also applied to the DDMR and the robot inputs $(v, \omega) \in \mathbb{R}^{2 \times 1}$ can be expressed as a function of the angular speeds $(\omega_L, \omega_R) \in \mathbb{R}^{2 \times 1}$ of the left and right wheels respectively as $v = r(\omega_L + \omega_R)/2$ and $\omega = r(\omega_L - \omega_R)/d$, where $r > 0$ is the wheels radius and $d > 0$ is a distance between their centers [16]. Then, the

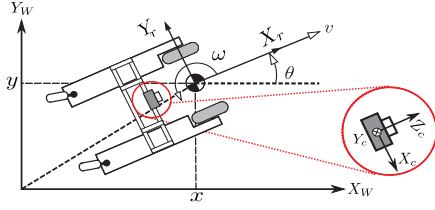


Fig. 2. Generalized coordinates $q = (x, y, \theta) \in \mathbb{R}^{3 \times 1}$ and the robot inputs $u = (v, \omega) \in \mathbb{R}^{2 \times 1}$ for a differential-drive mobile robot.

robot velocity $V_r = (v, 0, 0, 0, 0, \omega) \in \mathbb{R}^{6 \times 1}$ can be related to the robot inputs $u = (v, \omega) \in \mathbb{R}^{2 \times 1}$ using the following relationship:

$$V_r = S u, \quad S = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T, \quad (2)$$

where $S \in \mathbb{R}^{6 \times 2}$ is a selection matrix.

B. Camera Model

Consider a pin-hole camera observing a line \mathcal{L} with a set of image features (\mathcal{P}, Θ) , as described in Section II. Then, let $p_c = (x_c, y_c, z_c) \in \mathbb{R}^{3 \times 1}$ be the coordinates of the image feature point \mathcal{P} expressed in the camera frame \mathcal{F}_c . The velocity of \mathcal{P} expressed in frame \mathcal{F}_c is given by $\dot{p}_c = -\omega_c \times p_c - v_c$, where $v_c \in \mathbb{R}^{3 \times 1}$ and $\omega_c \in \mathbb{R}^{3 \times 1}$ are respectively the linear and angular velocities of the camera expressed in the world frame \mathcal{F}_w [17]. The image feature point \mathcal{P} can be expressed in the image frame \mathcal{F}_v with its projected coordinates $(x_v, y_v) \in \mathbb{R}^{2 \times 1}$ using the following relationship:

$$\begin{bmatrix} x_v \\ y_v \end{bmatrix} = \frac{f}{z_c} \begin{bmatrix} \alpha_x & 0 \\ 0 & \alpha_y \end{bmatrix} \begin{bmatrix} x_c \\ y_c \end{bmatrix} + \begin{bmatrix} x_{v0} \\ y_{v0} \end{bmatrix}, \quad (3)$$

where $(x_{v0}, y_{v0}) \in \mathbb{R}^{2 \times 1}$ is the camera principal point, $f > 0$ is the focal length of the camera lens, and $\alpha_x, \alpha_y > 0$ are the scaling factors. A line feature \mathcal{L} can be parameterized using a pair (ρ, θ_v) described by $y_v = -x_v \tan(\theta_v) + \rho / \cos(\theta_v)$ with $\rho \in [-\rho_{min}, \rho_{max}]$ being the perpendicular distance between the origin of the image frame \mathcal{F}_v and the line. The rate of change of the image features vector $s = (x_v, y_v, \theta_v) \in \mathbb{R}^{3 \times 1}$ is related to the camera spatial velocity $V_c = (v_c, \omega_c) \in \mathbb{R}^{6 \times 1}$ by [17]:

$$\dot{s} = L_s V_c + \xi_s, \quad (4)$$

where $L_s = [L_x \ L_y \ L_\theta]^T \in \mathbb{R}^{3 \times 6}$ is the so-called interaction matrix and $\xi_s = (\xi_x, \xi_y, \xi_\theta) \in \mathbb{R}^{3 \times 1}$ denotes the uncertain motion of the image feature, which can be considered as a bounded perturbation. In general, this disturbance term must be estimated to compensate for the target motion or it can be neglected when the robot is moving with low velocities and slow accelerations [17]. Our solution consists in designing a robust controller that explicitly deals with model inaccuracies and trajectory perturbations in the image space caused by high driving velocities and uneven terrains.

C. Camera-Robot System Model

The camera velocity $V_c = (v_c, \omega_c) \in \mathbb{R}^{6 \times 1}$ is related to the robot velocity $V_r \in \mathbb{R}^{6 \times 1}$ by using the adjoint matrix

$A_{cr} \in \mathbb{R}^{6 \times 6}$ as:

$$V_c = A_{cr} V_r, \quad A_{cr} = \begin{bmatrix} R_{cr} & [t_{cr}]_\times R_{cr} \\ 0_{3 \times 3} & R_{cr} \end{bmatrix}, \quad (5)$$

with $t_{cr} = (0, t_z, -t_x) \in \mathbb{R}^{3 \times 1}$ being the translation vector from the origin of the robot frame \mathcal{F}_r to the origin of the camera frame \mathcal{F}_c and $[t_{cr}]_\times \in \mathbb{R}^{3 \times 3}$ being a skew-symmetric matrix. The rotation matrix $R_{cr}(\psi) \in \mathbb{SO}(3)$ of the robot frame \mathcal{F}_r with respect to the camera frame \mathcal{F}_c is computed as $R_{cr}(\psi) = R_x(\psi) R_y(\pi/2) R_x(-\pi/2)$. The robot inputs $u = (v, \omega) \in \mathbb{R}^{2 \times 1}$ can be related to the camera velocity $V_c = (v_c, \omega_c) \in \mathbb{R}^{6 \times 1}$ by replacing Eq. (2) into Eq. (5) to obtain $V_c = T_{cr} u$ with $T_{cr} = A_{cr} S$. Then, substituting V_c into Eq. (4) gives the following IBVS system:

$$\dot{s} = L_s T_{cr} u + \xi_s, \quad (6)$$

where $T_{cr} = [T_v \ T_\omega] \in \mathbb{R}^{6 \times 2}$ can be split in two parts, with $T_v \in \mathbb{R}^{6 \times 1}$ and $T_\omega \in \mathbb{R}^{6 \times 1}$ being its first and the second columns respectively.

IV. IMAGE-BASED VISUAL SERVOING APPROACH

In the row crop following problem, it is possible to identify two common practical situations based on the robot location regarding the crop rows: (i) outer, when the robot is outside the crop rows; (ii) inner, when the robot is inside the crop rows. In [8], the authors have developed an IBVS approach using the analysis of the location of the image feature point \mathcal{P} in the image plane. Then, the proposed IBVS strategy is divided into two parts or primitives: a column controller and a row controller. In the next subsections, we briefly present the IBVS control design and a sequence of images representing the control action for both column and row cases. Notice that, in Figures 3 and 4, the red line denotes the current crop row orientation whereas the blue line denotes the desired path tangent orientation. Both angles are taken to be positive if the rotation of the lines is made counter-clockwise about y_v -axis of the image frame \mathcal{F}_v .

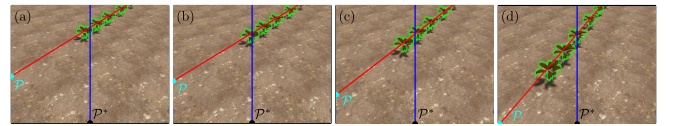


Fig. 3. An application example of the IBVS-CC approach.

A. IBVS Column Controller (IBVS-CC)

The IBVS-CC approach is applied for reaching the crop row or during maneuvers at the headlands, when the image feature point \mathcal{P} is located at the right (or left) pixel column of the image plane \mathcal{I} . Here, we assume that the control goal is to drive the current image coordinates (y_v, θ_v) to the desired values $y_v^* = H/2$ and $\theta_v^* = 0$, under the path following constraint $x_v^* = \pm W/2$ and $\dot{x}_v = 0$. An application example of the IBVS-CC approach is depicted in Fig. 3(a)-(d), where it can be observed a sequence of motions in which the robot pose is successfully controlled. In such a case, the current \mathcal{P} moves to the bottom corner of the image plane ($y_v = H/2$) and, then, the next step is to apply the IBVS row controller so that \mathcal{P} can achieve the desired \mathcal{P}^* located

at the bottom center of the image plane. Due to the velocity constraint $\dot{x}_v = 0$, the second and third rows of Eq. (6) can be selected, and the IBVS system is given by:

$$\dot{s}_c = \begin{bmatrix} \dot{y}_v \\ \dot{\theta}_v \end{bmatrix} = J_{v_c} v + J_{\omega_c} \omega + \xi_{s_c}, \quad (7)$$

where $\xi_{s_c} \in \mathbb{R}^{2 \times 1}$ is a perturbation term, with $J_{v_c} \in \mathbb{R}^{2 \times 1}$ and $J_{\omega_c} \in \mathbb{R}^{2 \times 1}$ being respectively the linear and angular Jacobian matrices that result from the multiplication of the matrices corresponding to L_s and T_{cr} , as shown below:

$$J_{v_c} = \begin{bmatrix} L_y \\ L_\theta \end{bmatrix} T_v, \quad J_{\omega_c} = \begin{bmatrix} L_y \\ L_\theta \end{bmatrix} T_\omega. \quad (8)$$

Then, the IBVS system, described in Eq. (7) with $\xi_{s_c} \approx 0$, can be controlled using the following control law:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} v_c \\ \omega_c \end{bmatrix}, \quad \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} = \begin{bmatrix} v_d \\ -J_{\omega_c}^\dagger (\Lambda_c e_c + J_{v_c} v_d) \end{bmatrix}, \quad (9)$$

where $\Lambda_c \in \mathbb{R}^{2 \times 2}$ is a positive definite gain matrix and $v_d \in \mathbb{R}$ is the constant desired linear velocity. The image features error $e_c = (e_y, e_\theta) \in \mathbb{R}^{2 \times 1}$ for the IBVS column controller is described by $e_y = y_v - y_v^*$, and $e_\theta = \theta_v - \theta_v^*$. The stability properties and convergence analysis for the IBVS-CC approach, given by Eq. (9), can be found in [8].

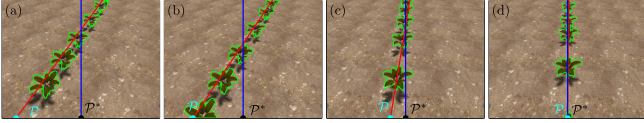


Fig. 4. An application example of IBVS-RC approach.

B. IBVS Row Controller (IBVS-RC)

The IBVS-RC approach is applied for row following task at the vicinity of the crop row, when the image feature point \mathcal{P} is located at the bottom (or top) pixel row of the image plane \mathcal{I} . Here, we assume that the control goal is to drive the current image coordinates (x_v, θ_v) to the desired values $x_v^* = 0$ and $\theta_v^* = 0$, under the path following constraint $y_v^* = H/2$ and $\dot{y}_v = 0$. An application example of the IBVS-RC approach is depicted in Fig. 4(a)-(d), where it can be observed a sequence of motions in which the robot pose is successfully controlled. In such a case, the current \mathcal{P} moves to the bottom center of the image plane ($x_v = 0$) and Θ becomes vertical ($\theta_v = 0$). Then, the row crop (red line) will align to the center of the image (blue line) and the offset between the lines will be zero. Due to the velocity constraint $\dot{y}_v = 0$, the first and third rows of Eq. (6) can be selected, and the IBVS system is given by:

$$\dot{s}_r = \begin{bmatrix} \dot{x}_v \\ \dot{\theta}_v \end{bmatrix} = J_{v_r} v + J_{\omega_r} \omega + \xi_{s_r}, \quad (10)$$

where $\xi_{s_r} \in \mathbb{R}^{2 \times 1}$ is a perturbation term, with $J_{v_r} \in \mathbb{R}^{2 \times 1}$ and $J_{\omega_r} \in \mathbb{R}^{2 \times 1}$ being respectively the linear and angular Jacobian matrices that result from the multiplication of the matrices corresponding to L_s and T_{cr} , as shown below:

$$J_{v_r} = \begin{bmatrix} L_x \\ L_\theta \end{bmatrix} T_v, \quad J_{\omega_r} = \begin{bmatrix} L_x \\ L_\theta \end{bmatrix} T_\omega. \quad (11)$$

Then, the IBVS system, described in Eq. (10) with $\xi_{s_r} \approx 0$, can be controlled using the following control law:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} v_r \\ \omega_r \end{bmatrix}, \quad \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} = \begin{bmatrix} v_d \\ -J_{\omega_r}^\dagger (\Lambda_r e_r + J_{v_r} v_d) \end{bmatrix}, \quad (12)$$

where $\Lambda_r \in \mathbb{R}^{2 \times 2}$ is a positive definite gain matrix. The image features error $e_r = (e_x, e_\theta) \in \mathbb{R}^{2 \times 1}$ for the IBVS row controller is described by $e_x = x_v - x_v^*$, and $e_\theta = \theta_v - \theta_v^*$. The stability properties and convergence analysis for the IBVS-RC approach, given by Eq. (12), can be found in [8].

V. ROBUST IMAGE-BASED VISUAL SERVOING

In this section, we shortly present the robust control design based on the SMC approach for the IBVS column and row controllers, introduced in the previous section. Here, let us introduce the following useful notation: $[\cdot]^\gamma = |\cdot|^\gamma \text{sgn}(\cdot)$ for $0 < \gamma \leq 1$, where $\text{sgn}(\cdot)$ is the sign function. Considering the model inaccuracies and trajectory perturbations in the designing of the robust IBVS approach, the linear and angular velocities, v and ω , can be written respectively as:

$$v = v_d, \quad \omega = \hat{J}_{\omega_i}^\dagger \nu_i, \quad (13)$$

where $\hat{J}_{\omega_i}^\dagger \in \mathbb{R}^{1 \times 2}$ is the pseudo-inverse of the nominal angular Jacobian matrix and $\nu_i \in \mathbb{R}^{2 \times 1}$ is the robust control action to be designed with $i = \{c, r\}$. Now, we obtain the following perturbed IBVS system:

$$\dot{s}_i = \nu_i + \eta_i, \quad (14)$$

where the disturbance term η_i is given by:

$$\eta_i = (I - J_{\omega_i} \hat{J}_{\omega_i}^\dagger) \nu_i + J_{v_i} v_d + \xi_{s_i}. \quad (15)$$

Notice that, the effect of the disturbance term η_i increases for high values of the desired driving velocity v_d and, consequently, the IBVS system may perform very poorly or become unstable. To cope with performance and instability issues caused by system's inaccuracies and perturbations, an alternative solution consists of using adaptive and robust control techniques due to their remarkable capability of dealing with parametric uncertainties and external disturbances [11].

A. Robust IBVS Column Controller (rIBVS-CC)

In the design of the rIBVS-CC approach, we consider the following robustness term $\tau_c \in \mathbb{R}^{2 \times 1}$ given by:

$$\tau_c = K_c [\sigma_c]^\gamma + \zeta_c, \quad \dot{\zeta}_c = -\mu_c \text{sgn}(\sigma_c), \quad (16)$$

where $K_c \in \mathbb{R}^{2 \times 2}$ and $\mu_c \in \mathbb{R}^{2 \times 2}$ are positive definite gain matrices, with $\sigma_c \in \mathbb{R}^{2 \times 1}$ being the sliding surface defined as $\sigma_c = e_c + \Gamma_c \int_0^t e_c(\tau) d\tau$, where $\Gamma_c \in \mathbb{R}^{2 \times 2}$ is a positive definite matrix. Hence, the rIBVS-CC approach can be implemented by applying Eq. (16) to Eq. (13) and adding a proportional term $\Lambda_c e_c$, as shown below:

$$\nu_c = \Lambda_c e_c + \tau_c. \quad (17)$$

B. Robust IBVS Row Controller (rIBVS-RC)

In the design of the rIBVS-RC approach, we consider the following robustness term $\tau_r \in \mathbb{R}^{2 \times 1}$ given by:

$$\tau_r = K_r [\sigma_r]^\gamma + \zeta_r, \quad \dot{\zeta}_r = -\mu_r \text{sgn}(\sigma_r), \quad (18)$$

where $K_{\mathcal{R}} \in \mathbb{R}^{2 \times 2}$ and $\mu_{\mathcal{R}} \in \mathbb{R}^{2 \times 2}$ are positive definite gain matrices, with $\sigma_{\mathcal{R}} \in \mathbb{R}^{2 \times 1}$ being the sliding surface defined as $\sigma_{\mathcal{R}} = e_{\mathcal{R}} + \Gamma_{\mathcal{R}} \int_0^t e_{\mathcal{R}}(\tau) d\tau$, where $\Gamma_{\mathcal{R}} \in \mathbb{R}^{2 \times 2}$ is a positive definite matrix. Hence, the rIBVS-RC approach can be implemented by applying Eq. (18) to Eq. (13) and adding a proportional term $\Lambda_{\mathcal{R}} e_{\mathcal{R}}$, as shown below:

$$\nu_{\mathcal{R}} = \Lambda_{\mathcal{R}} e_{\mathcal{R}} + \tau_{\mathcal{R}}. \quad (19)$$

Now, we can state the following theorem for establishing the stability properties of the rIBVS approach:

Theorem 1: Consider the IBVS system defined by Eq. (14) under the disturbance term given by Eq. (15), assumed to be bounded. Under the robust IBVS column and row controllers given by Eq. (17) and Eq. (19) respectively, the following properties hold: (i) all systems signals are bounded; (ii) $\lim_{t \rightarrow \infty} \sigma_c(t), \sigma_{\mathcal{R}}(t) = 0$; (iii) $\lim_{t \rightarrow \infty} e_x(t), e_{\theta}(t) = 0$; (iv) $\lim_{t \rightarrow \infty} e_y(t), e_{\theta}(t) = 0$.

Proof 1: For a proof, please see the Appendix I. ■

VI. 3D COMPUTER SIMULATIONS

In this section, we present 3D computer simulations and case studies with the rIBVS approach developed for autonomous visual navigation with a WMR in row crop fields. An example of a 3D agricultural environment with non-ideal conditions of plants and terrains was created in the ROS-Gazebo platform, as shown in Fig. 5. In real-world scenarios, row crop fields occur in rough and uneven terrains, and plants are sparsely distributed. The WMR used for autonomous navigation tests was the SoyBot [5], a DDMR equipped with a monocular camera attached at the top of its mechanical structure. The robot and the world (i.e., the crop field) models are described in an SDF file, which contains all their physical and geometrical characteristics. The visual and collision geometry of the robot are given in STL format developed by using *SolidWorks* CAD. Moreover, the camera and motor plugins are also given in the SDF file, that have the function of sending their dates to the user via ROS topic. The simulated field is created using a tool called a *heightmap*, which converts a 2D grayscale image into a 3D RGB format. This 3D model can be easily imported into the ROS-Gazebo platform [18]. To emulate the crop plants, we use a simple 3D model in STL format, called *acanthus*.

Here, we present the parameters of the camera-robot system used in the simulated experiments. The dimensions of the robot, that is, its width, length and height, are given respectively by $(W_r, L_r, H_r) = (1.2, 1.9, 2.0) m$. The diameter of the front wheels is $d_f = 40 cm$, and the castor wheels located at the robot back have $d_b = 20 cm$ of diameter. More detailed information about the SoyBot robot can be found in [5]. The camera used in the simulation was similar to a Logitech C270 HD webcam (720p, 30fps), with a focal length of $f = 8 mm$, and scale factors of $\alpha_x = 69.4 pixel mm^{-1}$ and $\alpha_y = 52.1 pixel mm^{-1}$ respectively. The RGB images captured by the camera have a dimension of 640×480 pixel resolution and are used for the navigation task after the line feature extraction.

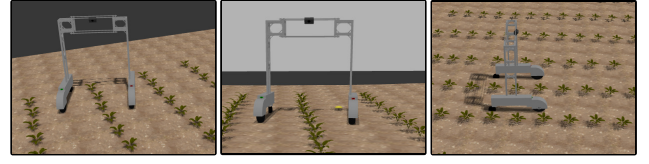


Fig. 5. The SoyBot robot navigating in the 3D agricultural environment created in the ROS-Gazebo platform, with sparse plants and rough terrains.

The camera is placed in vertical and horizontal coordinates, relative to the robot's center of mass, given by $(t_x, t_y, t_z) = (0.16, 0.0, 1.4) m$, with an offset tilt angle of $\psi = \pi/4 rad$. The communication between the sensors and the plugins was implemented using the Robot Operating System (ROS), release Kinetic Kame running on the Linux OS Ubuntu 16.04 LTS using a Intel Core i5-8250U 1.6GHz, 8GB DDR4 RAM. The ROS nodes for implementing the autonomous navigation task were developed in Python with OpenCV libraries. In the simulation tests, we considered a level of inaccuracy of 20% in the intrinsic and extrinsic camera calibration parameters. To verify the robustness properties of the proposed controller, we also include measurement noise in the image feature and unmodeled dynamics with high- or low-frequency to emulate the effects of actuator dynamics and Coulomb friction, respectively. In this case, the unmodeled dynamics can be represented by the transfer function $D(s) = c/(\lambda s + 1)$, where $c = 1$ and $\lambda \in [0.1, 0.5]$.

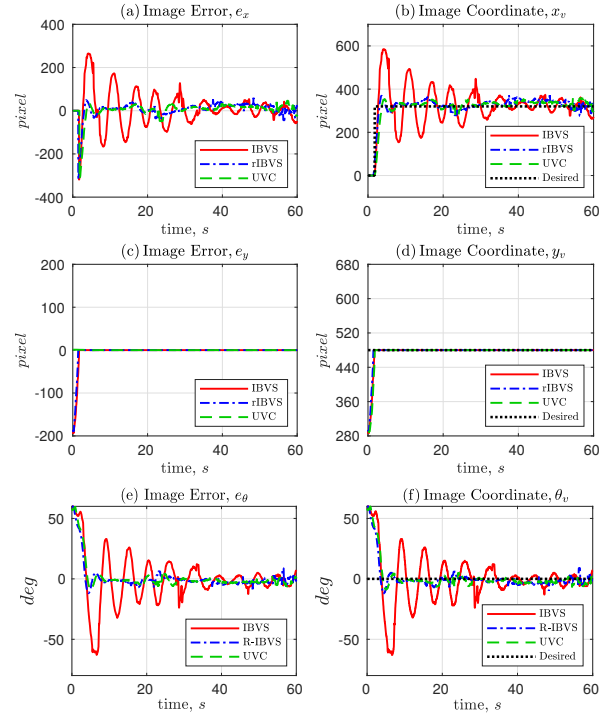


Fig. 6. Comparative study among the IBVS, rIBVS and UVC approaches: (a)-(b) image error, e_x and image coordinate, x_v ; (c)-(d) image error, e_y and image coordinate, y_v ; (e)-(f) image error, e_{θ} and image coordinate, θ_v .

Here, we will present a comparative study among the classic IBVS controller, given by Eq. (9) and Eq. (12), the rIBVS controller, given by Eq. (17) and Eq. (19) and the IBVS approach based on the unit vector control (UVC) law

described in [11]. We also consider a high value for the robot driving velocity, as $v_d = 1.5 \text{ m s}^{-1}$, to evaluate the performance of the IBVS controllers for the cases where the robot needs to move faster along the row crop field, relaxing the low-speed assumption used in [8]. In the simulations, the following values for the control parameters were used: $\Lambda_c = \text{diag}(0.3, 0.6)$ and $\Lambda_r = \text{diag}(2.5, 5.0)$; $K_c = 0.2 I$ and $K_r = 0.02 I$. Other parameters are: a sampling time of $h = 0.01 \text{ s}$ ($\approx 30 \text{ Hz}$) for a simulation time of $T_{max} = 60 \text{ s}$.

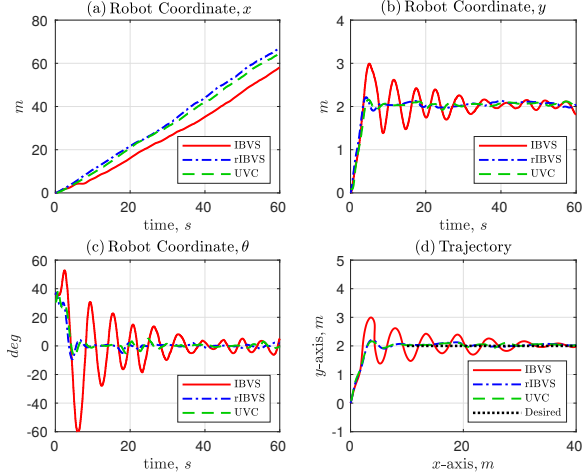


Fig. 7. Comparative study among the IBVS, rIBVS and UVC approaches: (a) robot coordinates, x_r , y_r and θ_r ; (b) trajectory traveled by the robot navigating along the crop row.

At the beginning of the experiments, the robot is located outside the row crop field. The column and row controllers are then activated to enable the robot to reach the crop row and align to it. Figures 6(a)-(b) and 6(c)-(d) depict the behavior over time of the image errors and image coordinates in x - and y -axis of the image frame \mathcal{F}_v respectively, whereas Fig. 6(e)-(f) depicts the image error e_θ and the image coordinate θ_v . We can observe that all controllers are able to drive the image coordinate y_v to the desired value $H/2$, and the robot successfully reaches the crop row. However, it is possible to notice the degraded performance obtained with the IBVS approach for image coordinates x_v and θ_v respectively. Therefore, the robot motion is impaired by unwanted oscillations during the row crop following, caused by external disturbances in the field and parametric uncertainties in the camera-robot system, which can damage neighboring plants and even the robot. On the other hand, as can be seen in Fig. 7(a)-(d), the rIBVS and UVC approaches are capable to deal with such trajectory perturbations and model inaccuracies, allowing the robot to carry out the autonomous navigation task safely and efficiently.

Finally, Fig. 8(a)-(b) shows the robot inputs (v, ω) and Fig. 8(c)-(d) depicts the robot wheel speeds (ω_L, ω_R), where it can be observed the remarkable performance of the rIBVS and UVC approaches in contrast to the oscillatory behavior of the classic IBVS approach. For comparative study purposes, we use two performance metrics as shown in Table I: (i) the root-mean-square error (RMSE) of the image feature errors e_x , e_y and e_θ ; (ii) the mean-absolute deviation (MAD)

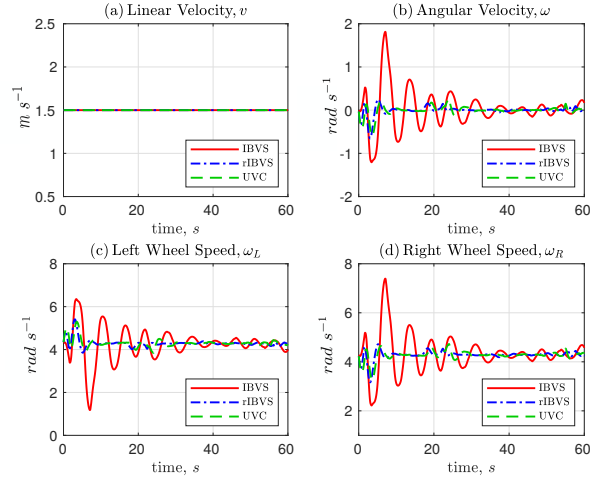


Fig. 8. Comparative study among the IBVS, rIBVS and UVC approaches: (a)-(b) linear and angular velocities; (c)-(d) left and right wheel speeds.

of the robot input ω , and the wheel speeds ω_L and ω_R . Analyzing the RMSE measure, we notice that the rIBVS approach shows the smallest values for the image features coordinates compared to IBVS and UVC approaches. Hence, the rIBVS approach presents a better transient behavior with fewer oscillations since the RMSE metrics penalize more the most significant errors. Moreover, Table I indicates that the rIBVS approach has lower average values for the control signals, which may be relevant for saving battery. Notice that, the rIBVS and UVC approaches can generate almost the same control effort for the robot wheels, which is less than the classic IBVS approach.

TABLE I
PERFORMANCE METRICS: RMSE AND MAD.

| RMSE | IBVS | rIBVS | UVC | unit |
|----------------------|-------|--------|--------|---------------------|
| x_v | 76.92 | 35.06 | 35.71 | pixel |
| y_v | 20.42 | 19.14 | 25.71 | pixel |
| θ_v | 0.321 | 0.198 | 0.214 | rad |
| MAD | IBVS | rIBVS | UVC | unit |
| ω | 0.261 | 0.0430 | 0.0533 | rad s ⁻¹ |
| ω_L, ω_R | 0.448 | 0.0737 | 0.0914 | rad s ⁻¹ |

VII. CONCLUDING REMARKS

In this work, we have proposed a novel robust image-based visual servoing (rIBVS) approach for wheeled mobile robots to perform row crop following using a single monocular camera. The key idea is to include a robustness term in the vision-based control law to deal with model inaccuracies caused by camera miscalibration and trajectory perturbations due to the robot's high driving velocities. A comparative study has been carried out considering three control strategies, the classic IBVS controller with row and column primitives, and two robust controllers based on the sliding mode control approach. The performance of the controllers has been evaluated using the RMSE and MAD values applied to the image errors and control signals. The stability and robustness properties of the rIBVS controllers have been analyzed using the Lyapunov stability theory. 3D computer simulations with a DDMR were carried out in

a row crop field built on the ROS-Gazebo platform¹. The virtual agricultural environment has been created to include irregularities in the terrain and sparse plants, emulating robot navigation for monitoring row crop fields.

APPENDIX I: STABILITY AND ROBUSTNESS ANALYSIS

Here, we present the stability and robustness analysis of the proposed rIBVS controllers. Then, the following assumptions for the perturbation term η_i given by Eq. (15) hold for all image features vector s_i for $i = \{c, \mathcal{R}\}$ as:

$$\sup_{t \geq 0} \|\xi_{si}\| < \Delta_{si} < \infty, \quad \forall \xi_{si}, \quad (20)$$

$$\|I - J_{\omega i} \hat{J}_{\omega i}^\dagger\| \leq \kappa \leq 1, \quad (21)$$

$$\|J_{vi} v_d\| \leq \beta < \infty. \quad (22)$$

The assumption given by Eq. (20) is satisfied because in Theorem 1 was assumed that ξ_{si} is bounded. To prove the assumption made in Eq. (21), we consider the boundedness of the norm of $\hat{J}_{\omega i}^\dagger$ given by the following inequalities:

$$0 < J_{mi} \leq \|\hat{J}_{\omega i}^\dagger\| \leq J_{Mi} < \infty, \quad (23)$$

where J_{mi} and J_{Mi} are the lower and upper bounds respectively. To satisfy the Eq. (21), we can choose $\hat{J}_{\omega i}$ as:

$$\hat{J}_{\omega i} = 2I / (J_{Mi} + J_{mi}). \quad (24)$$

Then, using Eq. (23) and Eq. (24) in the assumption given by Eq. (21), the following inequality can be found:

$$\|\hat{J}_{\omega i}^\dagger \hat{J}_{\omega i} - I\| \leq (J_{Mi} - J_{mi}) / (J_{Mi} + J_{mi}) = \kappa \leq 1. \quad (25)$$

Finally, the last assumption given by Eq. (22) is satisfied, since the parameters that define J_{vi} and the velocity v_d are also assumed to be bounded. For the stabilization of the rIBVS system in closed-loop (i.e., the regulation problem), the image error equation is taken from Eq. (14) as:

$$\dot{s}_i = \dot{e}_i = \nu_i + \eta_i, \quad (26)$$

where $e_i \in \mathbb{R}^2$ is a generic image error which can be defined in terms of image features errors e_c and $e_{\mathcal{R}}$. Now, to demonstrate the stability and robustness properties of the rIBVS system, we choose the Lyapunov function candidate: $2V_i(\sigma_i) = \sigma_i^\top \sigma_i$, which must satisfies the following properties: (P1) $V_i(\sigma_i) > 0$; (P2) $\dot{V}_i(\sigma_i) = \sigma_i^\top \dot{\sigma}_i \leq -\alpha |\sigma_i|$, where α is a strictly positive constant. Notice that, $\sigma_i \in \mathbb{R}^2$ is a generic sliding surface which can be defined in terms of $\sigma_{\mathcal{R}}$ and σ_c . Now, taking the time-derivative of $V(\sigma_i)$ and using Eq. (26), we obtain: $\dot{V}_i(\sigma_i) = \sigma_i^\top (\nu_i + \eta_i)$. Then, by choosing $\nu_i = -\Lambda \sigma_i - K_i [\sigma_i]^\gamma$, with $\Lambda_i = \Lambda_i^\top > 0$, yields $\dot{V}_i(\sigma_i) = -\sigma_i^\top \Lambda_i \sigma_i + \sigma_i^\top (\eta_i - K_i [\sigma_i]^\gamma)$, which gives $\sigma_i^\top (\eta_i - K_i [\sigma_i]^\gamma) \leq \|\sigma_i\| (\|\eta_i\| - K_i)$. To fulfill the sliding condition given by Property (P2), which makes the sliding surface an invariant set, the robustness gain matrix K_i must satisfy: $K_i \geq \|\eta_i\|$ for $\forall s_i, s_{di}$, which, in view of Eq. (20)-Eq. (22), implies that $\|\eta_i\| \leq \|J_{\omega i} \hat{J}_{\omega i} - I\| (\|\Lambda_i\| \|e_i\| + \|\tau_i\|) + \|J_{vi} v_d\| + \|\xi_{si}\|$. Then, setting $K_i \geq [\beta + \Delta_{si} + \kappa (\|\Lambda_i\| \|e_i\|)] / (1 - \kappa)$ results in $\dot{V}_i(\sigma_i) = -\sigma_i^\top \Lambda_i \sigma_i + \sigma_i^\top (\eta_i - K [\sigma_i]^\gamma) < 0$, which implies

that $\dot{V}_i(\sigma_i)$ is negative definite for all $\sigma_i \neq 0$. Therefore, we conclude that $\dot{V}_i(\sigma_i) \in \mathcal{L}_\infty$ and $\sigma_i, e_i \in \mathcal{L}_\infty$ and, consequently, $\lim_{t \rightarrow \infty} \sigma_i(t), e_i(t) = 0$ [16]. Notice that, from Eq. (13), we can also conclude that $\omega \in \mathcal{L}_\infty$ provided that $\nu_i \in \mathcal{L}_\infty$, which proves the property (i) from Theorem 1. Finally, we can apply these results to the rIBVS system and conclude that $\sigma_c, \sigma_{\mathcal{R}} \rightarrow 0$ and $e_x, e_y, e_\theta \rightarrow 0$, which prove the properties (ii), (iii) and (iv) from Theorem 1. ■

REFERENCES

- [1] A. Bechar and C. Vigneault, "Agricultural Robots for Field Operations: Concepts and Components," *Biosystems Engineering*, vol. 149, pp. 94–111, 2016.
- [2] T. D. Le, V. R. Ponnambalam, J. G. O. Gjevestad, and P. J. From, "A Low-cost and Efficient Autonomous Row-following Robot for Food Production in Polytunnels," *Journal of Field Robotics*, vol. 37, no. 2, pp. 309–321, 2020.
- [3] X. Gao, J. Li, L. Fan, Q. Zhou, K. Yin, J. Wang, C. Song, L. Huang, and Z. Wang, "Review of Wheeled Mobile Robots' Navigation Problems and Application Prospects in Agriculture," *IEEE Access*, vol. 6, pp. 49 248–49 268, 2018.
- [4] S. A. Fennimore and M. Cutulle, "Robotic Weeders Can Improve Weed Control Options for Specialty Crops," *Pest Management Science*, vol. 75, no. 7, pp. 1767–1774, 2019.
- [5] W. S. Barbosa, A. I. S. Oliveira, G. B. P. Barbosa, A. C. Leite, K. T. Figueiredo, M. M. B. R. Vellasco, and W. Caarls, "Design and Development of an Autonomous Mobile Robot for Inspection of Soy and Cotton Crops," in *12th International Conference on Developments in eSystems Engineering*, 2019, pp. 557–562.
- [6] L. Grimstad and P. J. From, "Thorvald II - a Modular and Reconfigurable Agricultural Robot," in *20th IFAC World Congress*, 2017, pp. 4588–4593.
- [7] F. Bonin-Font, A. Ortiz, and G. Oliver, "Visual Navigation for Mobile Robots: A Survey," *Journal of Intelligent and Robotic Systems*, vol. 53, pp. 263–298, 2008.
- [8] A. Cherubini, F. Chaumette, and G. Oriolo, "Visual Servoing for Path Reaching with Nonholonomic Robots," *Robotica*, vol. 29, no. 7, p. 1037–1048, 2011.
- [9] A. Ahmadi, L. Nardi, N. Chebrolu, and C. Stachniss, "Visual Servoing-based Navigation for Monitoring Row-Crop Fields," in *IEEE International Conference on Robotics and Automation*, 2020, pp. 4920–4926.
- [10] H. M. Becerra, G. López-Nicolás, and C. Sagüés, "A Sliding-mode-control Law for Mobile Robots based on Epipolar Visual Servoing from Three Views," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 175–183, 2011.
- [11] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, *Sliding Mode Control and Observation*. Springer Science+Business Media, 2014.
- [12] M. Thomas, B. Bandyopadhyay, and L. Vachhani, "Finite-time Posture Stabilization of the Unicycle Mobile Robot using only Position Information: A Discrete-time Sliding Mode Approach," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 6, pp. 1990–2006, 2019.
- [13] J. A. Moreno and M. Osorio, "Strict Lyapunov Functions for the Super-twisting Algorithm," *IEEE Transactions on Automatic Control*, vol. 57, no. 4, pp. 1035–1040, 2012.
- [14] H. M. Becerra, J.-B. Hayet, and C. Sagüés, "A Single Visual-servo Controller of Mobile Robots with Super-twisting Control," *Robotics and Autonomous Systems*, vol. 62, no. 11, pp. 1623–1635, 2014.
- [15] W. Winterhalter, F. V. Fleckenstein, C. Dornhege, and W. Burgard, "Crop Row Detection on Tiny Plants with the Pattern Hough Transform," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3394–3401, 2018.
- [16] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer-Verlag London, 2009.
- [17] F. Chaumette, S. Hutchinson, and P. Corke, "Visual Servoing," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer International Publishing, 2016, pp. 841–866.
- [18] B. Abbyasov, R. Lavrenov, A. Zakiev, K. Yakovlev, M. Svinin, and E. Magid, "Automatic Tool for Gazebo World Construction: from a Grayscale Image to a 3D Solid Model," in *IEEE International Conference on Robotics and Automation*, 2020, pp. 7226–7232.

¹The accompanying video clip for the autonomous row crop following can be found at the link <https://youtu.be/nXVUzyDoKQ0>.