

Adaptive End-effector Pose Control for Tomato Harvesting Robots

Dong Wang, Yongxiang Dong, Jie Lian, Dongbing Gu

Abstract—This paper investigates the development of a tomato-harvesting robot operating on a smart farm and primarily studies the reachable pose of tomatoes in the non-dexterous workspace of manipulator. The end-effector can only reach the tomatoes with reachable poses when the tomatoes is within the non-dexterous workspace. If the grasping posture is not reachable, it will lead to grasping failure. An adaptive end-effector pose control method based on a genetic algorithm(GA) is proposed to find a reachable pose. The inverse kinematic solution based on analysis method of the manipulator is analyzed and the objective function of whether the manipulator has a solution or not is obtained. The grasping pose is set as an individual owing to the position of the tomatoes is fixed and the grasping pose is variable. The GA is used to solve until a pose that can make the inverse kinematics have a solution is generated. This pose is the reachable grasping pose of the tomato at this position. The quintic interpolation polynomial is used to plan the trajectory to avoid damage to tomatoes owing to fast approaching speed and a distance based background filtering method is proposed. Experiments were performed to verify the effectiveness of the proposed method. The radius of the workspace of the UR3e manipulator with the end-effector increased from 550 to 800 mm and the grasping rang expanded by 208%. The harvesting success rate using the adaptive end-effector pose control method and trajectory planning method was 88%. The cycle of harvesting a tomato was 20s. The experimental results indicated that the proposed tomato-recognition and end-effector pose control method are feasible and effective.

Index Terms—Harvesting robot, manipulator, end-effector pose control, non-dexterous workspace.

I. INTRODUCTION

Tomatoes have a large demand and output all over the world as a vegetable with high nutritional values. The production of tomatoes in the world increased from 124 million tons to 177 million tons per year during 2003-2017. According to the data on the overall labor distribution of tomato cultivation, the highly repetitive task of harvesting accounts for 25%–40% of the total labor (Callejón-Ferre et al., 2015). Using large-scale agricultural harvesting devices for batch harvesting is difficult owing to the soft and fragile characteristics of ripe tomatoes. Hence, it is necessary to develop harvesting robots with autonomous perception and precise grasping capabilities

to reduce the harvesting costs and improve the harvesting efficiency. Planting is more standardized with the development of smart farms and plant factories making it possible to commercialize harvesting robots.

A robot for greenhouse tomato harvesting was developed(Lili et al., 2017). The built robot consisted of a four-wheel independent steering system, a five-degree-of-freedom (5-DOF) harvesting manipulator, and a binocular stereo vision steering system. It used the Otsu algorithm and an elliptical template method for tomato recognition and the harvesting success rate was approximately 86%. Ye et al. (Ye et al., 2021) developed a litchi harvesting robot, which used a particle swarm optimization algorithm to plan the obstacle avoidance trajectory of a 6-DOF manipulator and an improved rapidly exploring random tree (RRT) algorithm to improve the path planning speed. Ling et al. (Ling et al., 2019) designed a dual-arm collaborative robot for tomato harvesting and used the information of point clouds to reconstruct the scene for the recognition and positioning of tomatoes. One manipulator grasps the tomato and the other manipulator cuts the tomato stem when the target tomato is positioned. The success rate of tomato recognition was over 96%, and the harvest success rate reached 87.5%. Other scholars also performed tomato-harvesting experiments in laboratory environments (Fujinaga et al., 2019; Feng et al., 2015; Ishii et al., 2021; G. Liu et al., 2020; Benavides et al., 2020).

The success rate of harvesting is lower in the actual environment. There are some problems such as background interference, obstacle avoidance of manipulator and tomatoes not in the workspace. Yaguchi et al. (Yaguchi et al., 2016) developed a tomato-harvesting robot that can be used outdoors. The robot directly exposed to sunlight, its harvesting speed is about 23 s/fruit, and the success rate is approximately 60%. Silwal et al. (Silwal et al., 2017) built a field apple-harvesting robot. The robot uses the self-designed end-effector and uses color and point cloud for apple recognition and positioning. The positioning time is 1.5 s/fruit, and the success rate of apple harvesting is 84%. Arad et al. (Arad et al., 2020) built a field sweet-pepper-harvesting robot. The robot uses a depth camera combined with a color and shape recognition algorithm to identify sweet peppers and stems with a LED lighting system. The robot uses a visual servo control method to harvest sweet peppers. The harvesting success rate is 61% and the harvesting time is 24 s/fruit. Other scholars also conducted research on harvesting robot for tomatoes, sweet peppers, and other fruits (Lehnert et al., 2020; Gongal et al., 2015; Hayashi et al., 2010; Wang et al., 2019; Bac et al., 2014; Barth et al., 2019; Zhao et al., 2016; Lin et al., 2020; McCool et al., 2016).

Dong Wang, Yongxiang Dong, Jie Lian, the School of Control Science and Engineering, Dalian University of Technology, 116024, Dalian, China. Email: dwang@dlut.edu.cn.

Dongbing Gu, the School of Computer Science and Electronic Engineering, University of Essex, Colchester, United Kingdom.

This work was supported in part by the National Key Research and Development Program of China under grant 2019YFE0197700, the National Natural Science Foundation of China under Grant 61973050 and Grant 62173061, by Liaoning Revitalization Talents Program under Grant XLYC2007010, and by Fundamental Research Funds for the Central Universities under Grant DUT20GJ209 and Grant DUT20JC14.

Many scholars studied the dexterous workspace and grasping pose of manipulators. Rauchfuss et al. (Rauchfuss & Yang, 2000) calculated the reachable area of each joint by studying the conversion relationship of the wrist joint of the manipulator and finally obtained the reachable area of the end-effector of the manipulator in non-dexterous workspace. However, they did not discuss what pose can be reached in the non-dexterous workspace. Cao et al. (Cao et al., 2018) studied the dexterity of a manipulator. They defined the dexterity of the manipulator when grasping an object inspired by the human pose when grasping objects. The robot obtained the pose with high dexterity when grasping the target object and finally grasped it. Their study did not distinguish the non-dexterous workspace of the manipulator and they did not study the scenario in which the manipulator could not achieve a grasp with some pose in the non-dexterous workspace.

The pose calculation differs based on different inverse solution methods of manipulator. The solution methods of manipulator inverse kinematics (IK) are divided into two categories: analytical and numerical method. Aristidou et al. (Aristidou & Lasenby, 2011) proposed forward and backward reaching IK, which uses a forward and backward iterative approach to determine each joint position via locating a point on a line. This method improves the calculation speed of IK and reduces the calculation consumption. The commonly used IK algorithm TRAC-IK was proposed by Beeson et al. (Beeson & Ames, 2015). They aimed at solving the problem of open-source Orocos Kinematics and Dynamics Library repeatedly exhibiting false-negative failures on various humanoid platforms using a set of sequential quadratic programming IK algorithms that use various quadratic error metrics and a combined algorithm that concurrently runs the best performing sequential quadratic programming algorithm and the improved inverse Jacobian implementation to improve the success rate of IK. Diankov et al. (Diankov, 2010) designed the IK algorithm IK FAST based on the analytical method. The angles of each joint of the manipulator were obtained by solving the pose and position matrix of the manipulator. All the solutions that could reach the target point were obtained, and the desired optimal solution was obtained through selection.

In summary, the current research on tomato-harvesting robots primarily focuses on tomato recognition and grasping tasks. In laboratory and other structured environments, the success rate of tomato detection and grasping is 80%–90%. However, the success rates of tomato recognition and harvesting in outdoor unstructured environments is lower than that, at approximately 80% and 60%, respectively. In outdoor environments, the effectiveness of tomato recognition is easily affected by light and background interference. It is necessary to develop some filters to eliminate this interference. Owing to the narrow harvesting space, only a small manipulator can be used for harvesting, which results in many fruits appearing in the non-dexterous workspace, such as the boundary of the manipulator. The reachable end-effector pose is required to grasp the fruit in the non-dexterous workspace. Achieving a good effect in non-dexterous workspace using existing manipulator solution methods is difficult. The IK solution based on numerical method cannot separate the position matrix

from the pose matrix. Thus, it cannot obtain a more accurate pose when obtaining an accurate position. The IK based on the analytical method cannot obtain an approximate solution when the tomatoes are in a non-dexterous workspace and no solution is available for the given grasping pose. This results in a large number of points not being reached by the manipulator in the non-dexterous workspace.

This paper describes the development of a tomato-harvesting robot with focus on end-effector pose control, tomato recognition, and manipulator trajectory planning methods. First, an adaptive end-effector pose control method is proposed to solve the problem of the end-effector having difficulty reaching the tomatoes in the non-dexterous workspace. Scholars' research on the dexterous workspace of the manipulator has focused on finding the dexterous workspace of the manipulator, and then giving up the non-dexterous workspace to operate the manipulator in the dexterous workspace in recent years. However, many tomatoes are located in the non-dexterous workspace of the harvesting robot. It is necessary to overcome the difficulty of grasping with the manipulator in the non-dexterous workspace to improve the harvesting range and success rate of the harvesting robot. For the tomatoes in the non-dexterous workspace where the IK cannot be obtained with the current grasping pose, GA is used to find its reachable pose to make the manipulator have the IK solution and complete the grasp. The equation solved by the IK analysis method of the manipulator is analyzed, and the fitness function of whether the manipulator has a IK solution or not is obtained. The pose matrix is used as the input, and the pose that can make the manipulator have the IK solution is obtained using the GA without changing the position matrix. The result from this algorithm improves the success rate of harvesting tomatoes in non-dexterous workspace to expand the grasping range of the harvesting robot and accordingly improve the harvesting success rate. Second, a background filtering method is proposed, which can filter out the background whose distance is larger than the workspace of the manipulator. Third, The quintic interpolation polynomial is used to plan the end-effector trajectory to avoid damage to tomatoes owing to fast approaching speed. The planning result can limit the speed of the end-effector in the last stage near the tomato to ensure that the end-effector can perform the grasping task slowly. Finally, experiments were performed to verify the effectiveness of the proposed methods.

II. TOMATO HARVESTING ROBOT HARDWARE AND SOFTWARE ARCHITECTURE

A. Hardware system

The hardware system of the robot consists of a mobile platform, a 6-DOF manipulator UR3e, an end-effector, a depth camera, a 3D Lidar, and an industrial computer (Fig. 1). Specifically, the mobile platform at the bottom of robot is a SongLing ground mobile robot, which can rotate in all directions and navigate using the 3D Lidar installed on top of the control cabinet. The Universal Robot UR3e manipulator has six degrees of freedom and it is installed on top of the control cabinet with a payload of 3 kg and a workspace

radius of 500 mm. Fig. 2 presents the kinematics model of the UR3e manipulator, which was developed using the modified Denavit-Hartenberg (D-H) method. For the convenience of calculation, tool link 7 and the length of wrist3 to tool link 7 are added. The D-H modeling parameters are presented in Table I. The end-effector of manipulator is a pneumatic control soft gripper with a maximum load of 500 g and a length of 200 mm. A RealSense D435i depth camera installed on the end-effector and a RGB camera installed on the control cabinet are combined for tomato recognition and positioning. The industrial computer uses Intel i5 8400 CPU with 4.0 GB of running memory to host the entire robot software system.

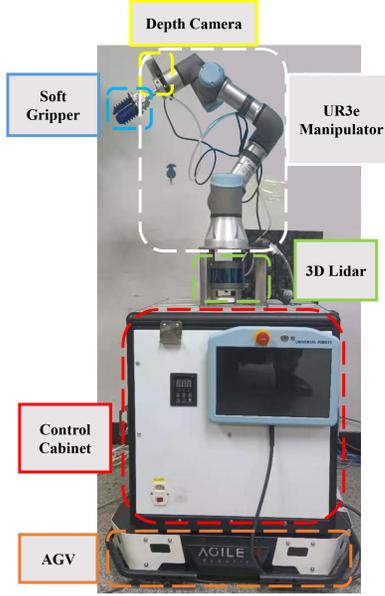


Fig. 1. Hardware system of tomato harvesting robot.

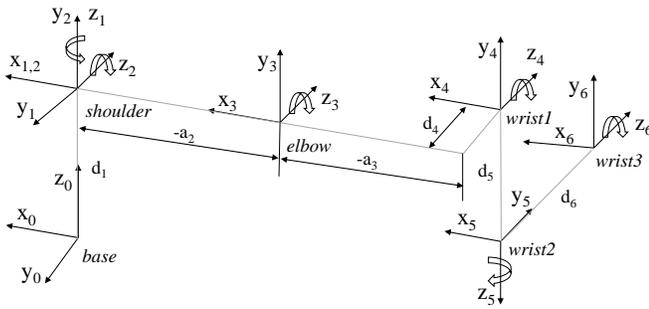


Fig. 2. Link coordination system of the robot.

B. Software framework

The software system architecture is shown in Fig. 3. The Robot Operating System (ROS) was installed in the Ubuntu 18.04 system of the industrial computer, which exchanges the information with various sensors and controllers. The industrial computer communicates with the depth camera, soft gripper, manipulator, and mobile robot using USB 3.0, an I/O

TABLE I
D-H PARAMETERS OF THE MANIPULATOR

Linkages i	Twist angles $a_i - 1/(^\circ)$	Length $a_i - 1/(mm)$	Offset of linkages $d_i/(mm)$	Angel range $\theta_i/(^\circ)$
1	0	0	151	$\theta_1 (0)$
2	90	0	0	$\theta_2 (-90)$
3	0	-243	0	$\theta_3 (0)$
4	0	-213	112	$\theta_4 (-90)$
5	90	0	85	$\theta_5 (0)$
6	-90	0	0	$\theta_6 (0)$
7	0	0	201	0

interface, and Ethernet to implement various tasks, including tomato recognition and positioning, harvesting manipulator control, end-effector control, and mobile robot control.

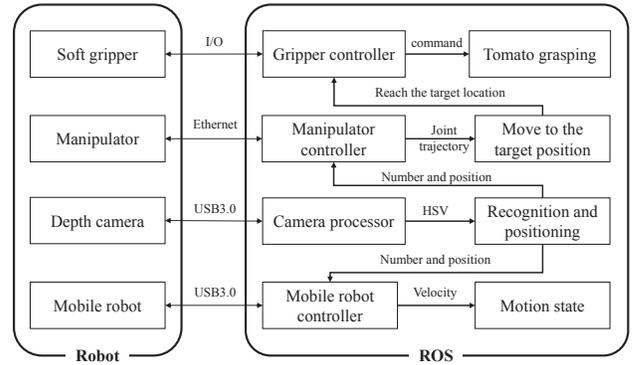


Fig. 3. Block diagram of tomato harvesting robot system.

C. Harvesting process

The harvesting process is shown in Fig. 4, which consists of four steps: moving and scanning, tomato recognition and positioning, manipulator action execution, and tomato grasping. In the first step, the harvesting robot uses the 3D Lidar and depth camera to navigate and build maps, and moves for tomato detection in the plant factory. The second step involves scanning the environment of the plant factory while the harvesting robot is moving to recognize the tomatoes. The depth information of a tomato center point is obtained by the depth camera, and the 3D coordinate of the tomato is determined based on the position of the tomato in the image. If the recognized tomato is determined to be in the workspace of the manipulator, the harvesting robot stops moving and releases the obtained coordinate of the tomato through the ROS node. In the third step, the harvesting manipulator reads the position of the tomato by subscribing to the ROS node, and converts the information from the camera coordinate frame to the manipulator coordinate frame. The manipulator uses the RRT algorithm to plan the path and uses the IK fast solver to obtain the joint angle of manipulator. The manipulator executes the obtained result and makes the gripper reach the tomato position. The final step closes the gripper to grasp the tomato and separates the stem by rotating the end joint of manipulator. **Pneumatic soft gripper is used to solve the grasping problem caused by different shapes and sizes of**

tomatoes. The selected soft grasper can grasp round objects between 4-8 cm and different shapes object such as eggs and bottles. The grasper completely overcomes individual differences in the same type of tomato.

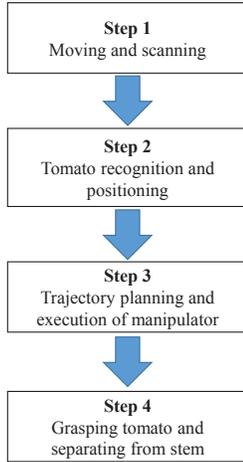


Fig. 4. Tomato harvesting process.

III. TOMATO RECOGNITION AND END-EFFECTOR POSE CONTROL METHODS

A. Tomato recognition method

Accurate recognition and positioning of tomatoes is the first important step for tomato harvesting robot. In this paper, the robot uses a RealSense D435i depth camera to scan the environment and perform the recognition and positioning tasks. The depth measurement range of the RealSense D435i depth camera is 0.2m-10m. The depth error is 1% of the measurement distance.

The recognition algorithm based on color segmentation can satisfy the requirements of tomato recognition as the colors of tomatoes and leaves differ significantly. The image is converted from the RGB color space to the HSV color space and the detection threshold range is set to H: 0-10, S: 43-255 and V: 46-255. The pixels within the set range are extracted and the tomato is positioned using a Hough circle fitting approach and the center position of tomatoes is obtained.

Background objects similar to the tomato colors may cause a false positive error for color-based recognition. A distance-based background filtering method is proposed to solve this problem. The UR3e manipulator used has a working radius of 500 mm. The soft gripper at the end of manipulator can extend the maximum workspace range to approximately 800 mm based on the depth information returned from the depth camera. The distance-based background filtering method can filter out the background pixels that are more than 850 mm away from the basis of manipulator. Fig. 5 shows the comparison between the original RGB image and the image after background filtering.

After background filtering, the color segmentation method is used to perform the tomato-recognition task. The coordinates of the recognized tomato center point are released through a



(a) Original RGB image (b) Image after background filtering

Fig. 5. Tomato background filtering.

ROS node, which provide a basis for the subsequent grasping task.

B. Manipulator workspace analysis

The manipulator has a primary workspace, also known as reachable workspace. Given a set of positions, if the end-effector of the manipulator has at least one pose to reach the position in the set, then the set is the reachable space of the manipulator. The secondary workspace, known as the dexterous workspace, is a subset of the primary workspace. Given a set of positions, if the end-effector of the manipulator has any pose solution to reach the position in the set, then the set is the secondary workspace of the manipulator. The area outside the dexterous workspace in the primary workspace is called the non-dexterous workspace (Siciliano & Khatib, 2016).

Owing to the narrow harvesting environment, only the small manipulator can be selected for harvesting. However, the workspace of a small manipulator is small, and the non-dexterous workspace accounts for a large proportion. As shown in Fig. 6, the dexterous workspace diameter of the UR3e manipulator is 1000 mm and the diameter of the primary workspace is 1146 mm(Universal ROBOTS, 2022). Since the workspace of the UR3e manipulator is spherical, the proportion of the non-dexterous workspace of the UR3e manipulator in the primary workspace can be calculated using (1),

$$o = \frac{r_m^3 - r_p^3}{r_m^3}, \quad (1)$$

where r_m is the radius of the primary workspace, r_p is the radius of the dexterous workspace. Using (1), it can observe that the area of the non-dexterous workspace accounts for 34% of the primary workspace.

If an end-effector is attached at the end of the manipulator, the proportion of non-dexterous workspace increases significantly. The dexterous and primary workspace of UR3e manipulator were calculated using MATLAB. The length of the end-effector is 200 mm. The dexterous workspace of manipulator was simulated using a fixed end-effector pose, and the primary workspace of the manipulator was simulated with a flexible pose. The Monte Carlo method is used to calculate the workspace, and randomly generated 100,000 points. As shown in Fig. 7, the diameter of dexterous workspace of UR3e manipulator reached 1100 mm, the primary workspace reached

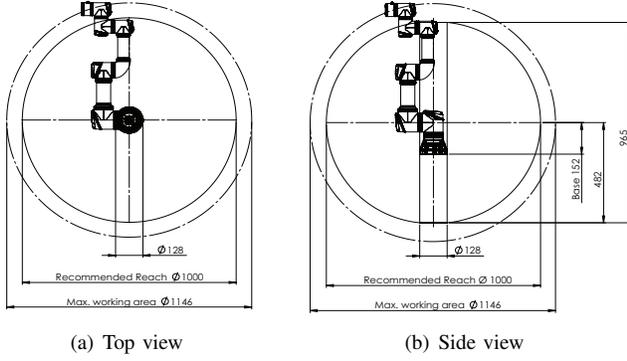
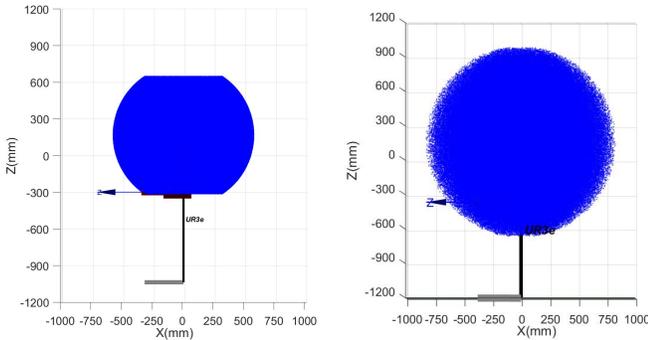


Fig. 6. Workspace of UR3e manipulator.

1600 mm, and the non-dexterous workspace accounted for 76% of the primary workspace after installing the end-effector. When the target point is in the non-dexterous workspace of the manipulator, the manipulator may not reach the target point with the desired pose, and may only reach the target point with a specific end-effector pose. This is why in the actual grasping process, the manipulator cannot reach the target point with the desired pose, but it can reach the target point by dragging the manipulator through teaching. If the UR3e manipulator can successfully reach all points in the non-dexterous workspace, its workspace can be expanded by 208% and the grasping range of the harvesting robot can be effectively improved.



(a) Workspace of UR3e obtained using a fixed pose (b) Workspace of UR3e obtained using a flexible pose

Fig. 7. Workspace of UR3e manipulator with different pose control method.

The harvesting robot must grasp non-spherical fruits such as cucumbers and eggplants in an end-effector pose perpendicular to the fruit. Unlike non-spherical fruits, spherical fruits such as tomatoes can be grasped and harvested in any pose. According to the grasping characteristics of spherical fruit, an adaptive end-effector pose control method is proposed. It involves calculating the end-effector pose that can cause the manipulator to reach the target point in the non-dexterous workspace to improve the harvesting range of the harvesting manipulator.

C. Initial grasping pose and interpolation points calculation

The appropriate initial grasping pose is preliminarily calculated using the position of the tomato and the current

end-effector position of the manipulator. The camera and end-effector of manipulator are calibrated using a nonlinear estimator method (Qiu et al., 2021). The rotation matrix of the manipulator end-effector with the depth camera can be obtained after calibration, and its rotation relationship can be added to ROS to establish the TF tree of the depth camera and manipulator, which can directly read the rotation matrix T_1 from the camera coordinate system *camera* to the manipulator base coordinate system *base* through ROS. TF conversion is performed through the rotation matrix T_1 to convert the tomato three-dimensional coordinates $a_0(x_{a0}, y_{a0}, z_{a0})$ obtained from the camera and the origin $g_0(0, 0, 0)$ of the tool coordinate system of the manipulator end-effector into the base coordinate system *base*. The tomato position $a(x_{a1}, y_{a1}, z_{a1})$ and $g(x_g, y_g, z_g)$ in the manipulator base coordinate system is calculated as

$$a = T_1 \times a_0, \quad (2)$$

$$g = T_1 \times t_0. \quad (3)$$

The distance from the tomato to the base of the manipulator is calculated to determine whether the target tomato center point is in the workspace of the manipulator,

$$l = \sqrt{x_{a1}^2 + y_{a1}^2 + z_{a1}^2}. \quad (4)$$

The threshold value of the distance is 850mm since the radius of UR3e workspace is 800mm after the installation of the soft gripper and to avoid the positioning error of the camera. If $l > 850$, the tomato is beyond the grasping range of the manipulator, and the tomato is abandoned. If $l < 850$, the tomato is within the grasping range of the manipulator. Subsequently, the grasping pose is calculated. The direction vector a_0 between the manipulator's end-effector coordinate system origin $g(x_g, y_g, z_g)$ and the tomato position $a(x_{a1}, y_{a1}, z_{a1})$ is calculated, and a_0 is unitized to obtain a unit vector a as

$$a_0 = [x_{a1} - x_g, y_{a1} - y_g, z_{a1} - z_g], \quad (5)$$

$$a = \frac{a_0}{|a_0|} = [x_a, y_a, z_a], \quad (6)$$

where the direction vector a is the desired pose of the harvesting manipulator's end-effector. The obtained direction vector a can only represent the z-axis direction of the desired end-effector pose. Additionally, its conversion relationship with the z-axis of the base coordinate system must be calculated. To make the pose of manipulator end-effector the same as the direction vector a , the *base* coordinate system is rotated such that the z-axis direction vector $z[0, 0, 1]$ is rotated as $a[x_a, y_a, z_a]$. The coordinate system transformation relationship is shown in Fig. 8.

To avoid the deadlock phenomenon of the manipulator, quaternion is needed to represent the rotation. The rotating quaternion q is represented by

$$q = \cos \frac{\theta}{2} + \sin \frac{\theta}{2} \mathbf{u}, \quad (7)$$

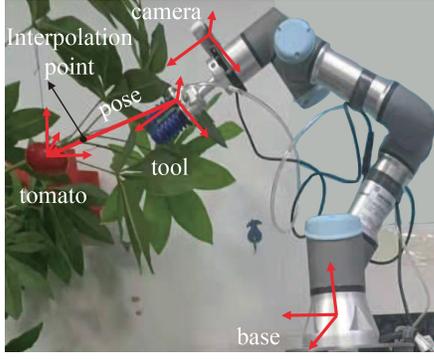


Fig. 8. Transformation relationship of coordinate system. The vector between the end-effector of the manipulator and the tomato is used as the grasping pose and an interpolation point is added at the end of the grasping vector to ensure that the manipulator grasps the tomato in the correct direction.

where θ is the angle between z and a . u is the direction vector perpendicular to z and a .

To facilitate the calculation of $\cos \frac{\theta}{2}$ and $\sin \frac{\theta}{2}$, let h be the intermediate vector at $\frac{\theta}{2}$ between z and a . Then h is

$$h = \frac{z + a}{|z + a|}. \quad (8)$$

Since u is the direction vector perpendicular to $z + a$,

$$a \cdot h = \cos \frac{\theta}{2}, \quad (9)$$

$$a \times h = \sin \frac{\theta}{2} u. \quad (10)$$

With the quaternion q_1 from which the z-axis of manipulator end-effector coordinate system rotates to the rotation vector a , the quaternion q_1 can be obtained by

$$q = a \cdot h + a \times h, \quad (11)$$

thus the rotation quaternion q is obtained, multiplying the manipulator end-effector coordinate frame z-axis vector z by the rotation quaternion q gets the manipulator end-effector pose p as

$$p = q \times z. \quad (12)$$

Finally, harvesting robot can determine the grasping pose and position of the end-effector by combining the position information of the tomato given by the camera.

D. Analysis of inverse kinematics based on analytical method

The obtained end-effector grasping pose is used for path planning. If the manipulator can successfully perform path planning, grasping is executed. However, when the tomato is in the non-dexterous workspace, there is often no IK solution with the initial grasping pose. As shown in Fig. 9, the spherical surface is used to simplify the grasping pose, and each vector from the center of the spherical to the point on the spherical surface is used as a grasping pose. When the tomato is located in the non-dexterous workspace of the manipulator, the grasping pose of the tomato is divided into two categories. One is the pose that can reach the center point of the tomatoes. The set of reachable poses is called reachable pose area, and the other is the unreachable pose area. When tomato is in different

positions of the manipulator, its reachable and unreachable pose areas are different. When the initial tomato grasping pose is in the unreachable pose region, a pose that can make the manipulator reach the target point must be determined. If this pose can be determined, the manipulator can grasp the tomato. The combination of analytical solution algorithm and GA is used to determine the reachable pose of the manipulator. The implementation process of this method is described in detail below.

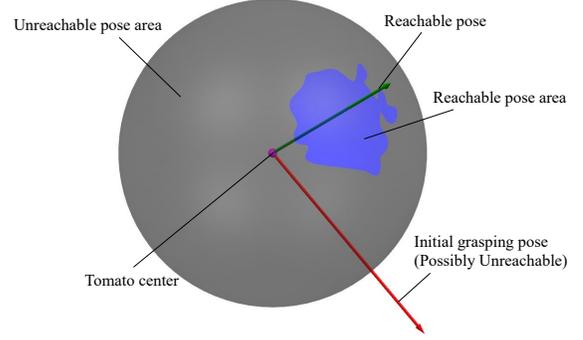


Fig. 9. The reachable pose area of tomatoes at different positions in manipulator coordinates is different. If the grasping pose obtained by relative position is not reachable, it is necessary to find a reachable pose to complete the grasping of tomatoes.

The path planning and end-effector control of the manipulator require the mapping from the Cartesian space coordinates of the end effector to the joint space of the manipulator. As shown in Fig. 10, the process of using all joint angles to obtain the position and pose matrix of the end-effector is called forward kinematics, and determining the angle of each joint is called IK when the position and pose matrix of the end-effector are known. The forward kinematics of the manipulator can be

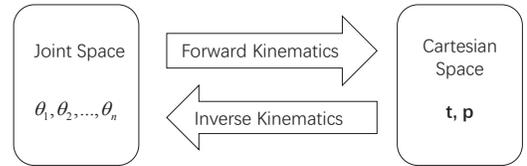


Fig. 10. Relationship between forward and inverse kinematics.

solved when the structural parameters and joint angles of the manipulator are known. The harvesting manipulator model is shown in Table 1 and Fig. 2. The conversion matrix between two adjacent links of the manipulator can be calculated using the D-H parameters in Table 1 and (13) as

$${}_{i-1}T_i = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & \alpha_{i-1} \\ S\theta_i C\alpha_{i-1} & C\theta_i C\alpha_{i-1} & -S\alpha_{i-1} & -d_i S\alpha_{i-1} \\ S\theta_i S\alpha_{i-1} & C\theta_i S\alpha_{i-1} & C\alpha_{i-1} & d_i C\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (13)$$

The structural parameters of the manipulator and (13) are combined to obtain the homogeneous transformation matrix ${}^0T_1 - {}^6T_7$, and the homogeneous transformation matrix ${}^0T_1 - {}^6T_7$ is multiplied to obtain the homogeneous transformation matrix

T_{tool} of the manipulator from the *base* to the *tool* coordinate system as

$$T_{tool} = {}^0_1T_2T_3T_4T_5T_6T_7T, \quad (14)$$

calculate 6_0T from 7_0T and 7_6T ,

$${}^6_0T = T_{tool} \times {}^6_7T^{-1} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (15)$$

where

$$\begin{aligned} n_x &= -S_6C_1S_{234} + S_1S_5C_6 + C_1C_5C_6C_{234}, \\ o_x &= -S_1S_5S_6 - C_1C_{234}C_5S_6 - C_1C_6S_{234}, \\ a_x &= S_1C_5 - S_5C_1C_{234}, \\ p_x &= d_4S_1 + d_5C_1S_{234} + a_2C_1C_2 + a_3C_1C_{23}, \\ n_y &= S_1C_{234}C_5C_6 - S_5C_1C_6 - S_1S_{234}S_6, \\ o_y &= -S_1S_6C_{234}C_5 + S_5S_6C_1 - S_1S_{234}C_6, \end{aligned}$$

$$\begin{aligned} a_y &= -S_1S_5C_{234} - C_1C_5, \\ p_y &= -d_4C_1 + d_5S_1S_{234} + a_2S_1C_2 + a_3S_1C_{23}, \\ n_z &= S_{234}C_5C_6 + S_6C_{234}, \\ o_z &= C_{234}C_6 - S_{234}S_6C_5, \\ a_z &= -S_{234}S_5, \\ p_z &= -d_5C_{234} + a_2S_2 + a_3S_{23} + d_6, \\ S_i &= \sin\theta_i, \\ C_i &= \cos\theta_i, \\ S_{23} &= \sin(\theta_2 + \theta_3), \\ C_{23} &= \cos(\theta_2 + \theta_3), \\ S_{234} &= \sin(\theta_2 + \theta_3 + \theta_4), \\ C_{234} &= \cos(\theta_2 + \theta_3 + \theta_4). \end{aligned}$$

6_1T is used to solve the equation to simplify the calculation of the IK of the manipulator to find the relationship between the the IK and the grasping pose. Solve θ_1 - θ_6 by making the terms on both sides of the equation 6_1T equal.

$${}^6_1T = \begin{bmatrix} C_{234}C_5C_6 - S_{234}S_6 & -C_{234}C_5S_6 - S_{234}C_6 & -C_{234}S_5 & a_3C_{23} + a_2C_2 + d_5S_{234} \\ -S_5C_6 & S_5S_6 & -C_5 & -d_4 \\ S_{234}C_5C_6 + C_{234}S_6 & C_{234}C_6 - S_{234}C_5S_6 & -S_{234}S_5 & a_3S_{23} + a_2S_2 - d_5C_{234} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (16)$$

$${}^6_1T = \begin{bmatrix} C_1 & S_1 & 0 & 0 \\ -S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & -d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (17)$$

A formula is proved to facilitate our next calculation. Suppose there is $-\sin(\theta)p_x + \cos(\theta)p_y = -d$, then $\theta = \text{atan2}(p_y, p_x) - \text{atan2}(-d, \pm\sqrt{p_x^2 + p_y^2 - d^2})$, where $\text{atan2}(x, y)$ is the azimuth from the origin to the point (x, y) . Prove, let $p_x = \rho \cos(\phi)$, $p_y = \rho \sin(\phi)$, where $\rho = \sqrt{p_x^2 + p_y^2}$, $\phi = \text{atan2}(p_y, p_x)$. It can be obtained by substituting into the original equation,

$$\begin{cases} \cos(\theta) \sin(\phi) - \sin(\theta) \cos(\phi) = -\frac{d}{\rho} \\ \sin(\phi - \theta) = -\frac{d}{\rho} \\ \cos(\phi - \theta) = \pm\sqrt{1 - \frac{d^2}{\rho^2}} \\ \phi - \theta = \text{atan2}\left(-\frac{d}{\rho}, \pm\sqrt{1 - \frac{d^2}{\rho^2}}\right) \end{cases}, \quad (18)$$

from (18),

$$\theta = \text{atan2}(p_y, p_x) - \text{atan2}(-d, \pm\sqrt{p_x^2 + p_y^2 - d^2}). \quad (19)$$

It can be obtained from (16) and (17),

$$-S_1p_x + C_1p_y = -d_4. \quad (20)$$

It can be obtained by substituting into (20),

$$\theta_1 = \text{atan2}(p_y, p_x) - \text{atan2}(-d_4, \pm\sqrt{p_x^2 + p_y^2 - d_4^2}), \quad (21)$$

for θ_1 to have a solution, $\sqrt{p_x^2 + p_y^2 - d_4^2} \geq 0$ is required. For θ_1 , p_x and p_y are the factors that determine whether there is a solution. They are only related to the position of the tomato,

not to the grasping pose. It can be obtained from the (16) and (17),

$$-S_1n_x + C_1n_y = -S_5C_6, \quad (22)$$

$$-S_1o_x + C_1o_y = S_5S_6, \quad (23)$$

$$-S_1a_x + C_1a_y = -C_5, \quad (24)$$

from (22), (23),

$$S_5 = \pm\sqrt{(-S_1n_x + C_1n_y)^2 + (-S_1o_x + C_1o_y)^2}, \quad (25)$$

$$S_6 = \frac{-S_1o_x + C_1o_y}{S_5}, \quad (26)$$

$$C_6 = \frac{S_1n_x - C_1n_y}{S_5}, \quad (27)$$

from (24),

$$\theta_5 = \text{atan2}(S_5, S_1a_x - C_1a_y), \quad (28)$$

from (25), (26),

$$\theta_6 = \text{atan2}\left(\frac{-S_1o_x + C_1o_y}{S_5}, \frac{S_1n_x - C_1n_y}{S_5}\right), \quad (29)$$

θ_5 and θ_6 has solution when θ_1 has a solution. Therefore, when $\sqrt{p_x^2 + p_y^2 - d_4^2} \geq 0$, the IK of the three joint angles θ_1 , θ_5 , θ_6 of the manipulator can be obtained. The IK of the remaining three joint angles are calculated to explore the relationship between the grasping pose and the IK of the manipulator. From (16), (17),

$$C_1a_x + S_1a_y = -C_{234}S_5, \quad (30)$$

$$a_z = -S_{234}S_5, \quad (31)$$

$$C_1p_x + S_1p_y = a_3C_{23} + a_2C_2 + d_5S_{234}, \quad (32)$$

$$p_z - d_1 = a_3S_{23} + a_2S_2 - d_5C_{234}, \quad (33)$$

from (30), (31),

$$\theta_{234} = \text{atan2}\left(-\frac{a_z}{S_5}, -\frac{C_1 a_x + S_1 a_y}{S_5}\right), \quad (34)$$

from (32), (33),

$$-AS_2 + BC_2 = C, \quad (35)$$

where, $A = -2B_2 a_2$, $B = 2B_1 a_2$, $C = B_1^2 + B_2^2 + a_2^2 - a_3^2$, $B_1 = C_1 p_x + S_1 p_y - d_5 S_{234}$, $B_2 = p_z - d_1 + d_5 C_{234}$. (35) has the same formula form as (19), so it can be obtained from (35) and (19),

$$\theta_2 = \text{atan2}(B, A) - \text{atan2}(C, \pm\sqrt{A^2 + B^2 - C^2}), \quad (36)$$

from (36), if $\sqrt{A^2 + B^2 - C^2} \geq 0$, θ_2 has a solution. Take θ_2 into (32), (33),

$$S_{23} = \frac{B_2 - a_2 S_2}{a_3}, \quad (37)$$

$$C_{23} = \frac{B_1 - a_2 C_2}{a_3}, \quad (38)$$

$$\begin{cases} \theta_{23} = \text{atan2}\left(\frac{B_2 - a_2 S_2}{a_3}, \frac{B_1 - a_2 C_2}{a_3}\right) \\ \theta_4 = \theta_{234} - \theta_{23} \\ \theta_3 = \theta_{23} - \theta_2 \end{cases} \quad (39)$$

All 6 joint angles of the IK of the manipulator have been obtained. From (36) - (39), $\sqrt{A^2 + B^2 - C^2} \geq 0$, θ_2 , θ_3 and θ_4 have solutions. Then this pose is the reachable pose of the manipulator at the target point. According to (21) - (39), $\sqrt{p_x^2 + p_y^2 - d_4^2}$ and $\sqrt{A^2 + B^2 - C^2}$ restrict the existence of inverse solution of the manipulator. When $\sqrt{p_x^2 + p_y^2 - d_4^2} \geq 0$ and $\sqrt{A^2 + B^2 - C^2} \geq 0$, the IK of all six joint angles of the manipulator can be obtained. It is only related to the position of the tomatoes and the manipulator, not the harvesting pose. Adjusting the grasping pose of the manipulator will not affect $\sqrt{p_x^2 + p_y^2 - d_4^2}$.

$\sqrt{A^2 + B^2 - C^2}$ contains the elements p_x , p_y , p_z in the position matrix and a_x , a_y , a_z , o_x , o_y in the pose matrix. This equation is not only related to the tomato position, but also related to the harvesting pose. This is a complex multi-variate problem. When the tomatoes are in the non-dexterous workspace of the manipulator, if the harvesting pose may not be reachable by the manipulator, $\sqrt{A^2 + B^2 - C^2} < 0$ occur. Then only 3 joint angles of the manipulator can be obtained and the remaining 3 have no solution. Therefore, GA is used to find the reachable grasping pose of the manipulator. The reachable grasping pose of the manipulator at this point can be found by obtaining the pose matrix that makes $\sqrt{A^2 + B^2 - C^2} \geq 0$.

E. Calculation of reachable pose based genetic algorithm

The three components of Euler angles of manipulator end-effector pose are coded into three genotypes respectively. The rotation Euler angle that *base* coordinate system z-axis rotates to the direction vector \mathbf{a} is calculated by

$$\begin{cases} \psi = \arctan\left(\frac{z_a}{y_a}\right) \\ \theta = \arctan\left(\frac{z_a}{x_a}\right) \\ \varphi = \arctan\left(\frac{y_a}{x_a}\right) \end{cases} \quad (40)$$

A group of Euler angles is represented as an individual, and different Euler angles represent different genotypes. The individual representation is expressed as

$$x = (\psi_n, \theta_n, \varphi_n). \quad (41)$$

To improve the convergence speed of the algorithm, the search space of end-effector pose is appropriately reduced. When the manipulator cannot reach the center point of the target tomato with the pose towards the target tomato, it must not reach from the back direction of the target tomato. The search space of the end-effector pose is limited to the hemisphere with the pose calculated in (40) as the center and the target tomato center as the sphere center. This is achieved by limiting the upper and lower limits of three genotypes. The search space for each genotype is expressed as

$$\begin{cases} \arctan\left(\frac{z_a}{y_a}\right) - \frac{\pi}{2} \leq \psi_i \leq \arctan\left(\frac{z_a}{y_a}\right) + \frac{\pi}{2} \\ \arctan\left(\frac{z_a}{x_a}\right) - \frac{\pi}{2} \leq \theta_i \leq \arctan\left(\frac{z_a}{x_a}\right) + \frac{\pi}{2} \\ \arctan\left(\frac{y_a}{x_a}\right) - \frac{\pi}{2} \leq \varphi_i \leq \arctan\left(\frac{y_a}{x_a}\right) + \frac{\pi}{2} \end{cases} \quad \forall i = 1, \dots, n.$$

After limiting the range of each genotype, the individual Euler angle composed of each genotype is combined with the position of the tomato center to obtain the position and pose matrix. Take the position and pose matrix into $\sqrt{A^2 + B^2 - C^2}$. From the above analysis, all 6 joint angles of the manipulator have solutions when $\sqrt{A^2 + B^2 - C^2} \geq 0$. Therefore, take $\sqrt{A^2 + B^2 - C^2}$ as the objective function, the larger the value of $\sqrt{A^2 + B^2 - C^2}$, the higher the fitness. The objective function and fitness of the algorithm are designed as

$$K = A^2 + B^2 - C^2, \quad (42)$$

$$\phi = K \quad (43)$$

The population size is set to 17, and the initialization population is evenly distributed on a 45 degree sphere centered on the calculated pose vector in (16). The center of the sphere is the center point of the target tomato. The initial individuals of the population are evenly sampled in each interval and coordinate plane of the 45 degree hemisphere. As shown in Fig. 11, the red vector is the calculated initial pose vector in (16), the blue vector is the initialization population vector, and the green sphere is the search space of the GA. A uniform distribution can avoid the gene aggregation of randomly distributed initial individuals and cause the population to converge more rapidly. The initial individuals of the population are expressed as

$$\begin{aligned} x_1 &= (\psi, \theta, \varphi), x_2 = \left(\psi + \frac{\pi}{8}, \theta, \varphi\right), x_3 = \left(\psi - \frac{\pi}{8}, \theta, \varphi\right), \\ x_4 &= \left(\psi + \frac{\pi}{4}, \theta, \varphi\right), \dots, x_{17} = \left(\psi + \frac{\pi}{4}, \theta + \frac{\pi}{4}, \varphi\right). \end{aligned}$$

Roulette wheel selection is used for individual selection, one-point crossover method is used for inheritance, and simple mutation method is used for gene variation and iteration is set to 2000. When the objective function $K \geq 0$, the cycle stopped and the obtained end-effector pose and the IK of 6

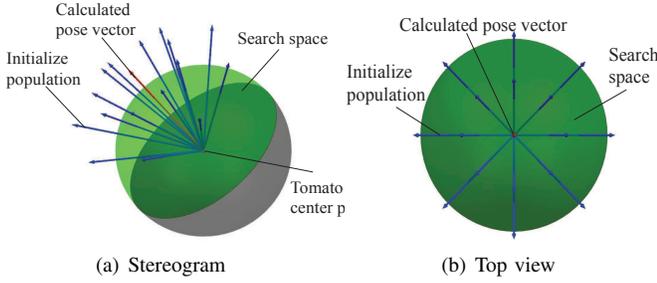


Fig. 11. The search area is the hemisphere centered on the grasping pose calculated from the relative position and 16 initial poses are generated as the initialized population centered on the grasping pose calculated from the relative position.

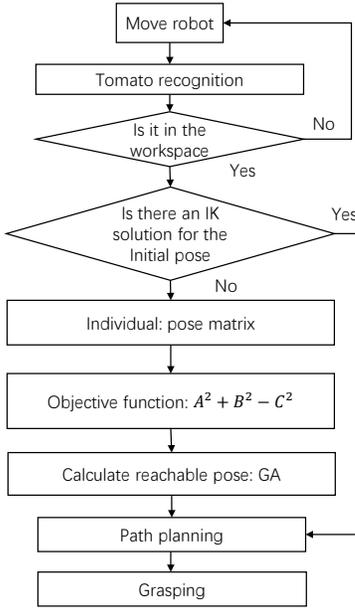


Fig. 12. Flowchart of the adaptive end-effector pose control algorithm.

joint angles are returned. The flowchart of the adaptive end-effector pose control algorithm is given in Fig. 12 and the algorithm is expressed as

Remark 1: Aiming at the problem that the manipulator has difficult grasping tomatoes in the non-dexterous workspace, an adaptive end-effector pose control method is proposed. For the target tomato that cannot be reached with the calculated initial end-effector pose, GA is used to determine the end-effector pose that can make the manipulator reach the target point. The adaptive end-effector pose control method can fully utilize the manipulator workspace, reach more grasping positions, and improve the success rate of tomato harvesting.

F. Grasping trajectory planning for harvesting manipulator

The tomato harvesting robot is equipped with a UR3e 6-DOF manipulator and a soft gripper to grasp tomatoes. During path planning of the manipulator, if no constraints are added, the planned path of manipulator may approach the tomato from any direction. However, the existence of the gripper determines that harvesting robot can only grasp a tomato from

Algorithm 1: Adaptive end-effector pose control algorithm

Input: \mathbf{a} , \mathbf{g} and r_m .

```

1 if  $|\mathbf{a}| < r_m$  then
2    $\mathbf{a}_0 = \mathbf{a} - \mathbf{g}$ ;
3    $\mathbf{p} = \frac{\mathbf{a}_0}{|\mathbf{a}_0|}$ ;
4   if  $K < 0$  then
5     Initialize P(i);
6      $i = 1$ ;
7     while (( $i < 2000$ ) or ( $K < 0$ ));
8     do
9       Evaluate and Sort P(i) by Fitness;
10      Select operation to P(i);
11      Crossover operation to P(i);
12      Mutation operation to P(i);
13       $P(i+1) = P(i)$ ;
14       $i = i + 1$ ;
15    end
16  end
17 end

```

the gripper-opening direction on the final distance close to tomato. Otherwise, the manipulator approaching the tomato from other directions will cause the gripper to collide with the tomato and fail to grasp it correctly. To overcome this challenge, a method based on adding interpolation point to constrain the manipulator is proposed to enable the planned path of manipulator to approach the tomato along the direction of the gripper opening and avoid collision between gripper and tomato.

The interpolation point is the point that manipulator end-effector must pass during path planning. Under the manipulator end-effector coordinate frame $tool$, the interpolation b is set at 100 mm in front of the tomato position $a(x_a, y_a, z_a)$ on end-effector pose p . In Section III.C, the pose p of the manipulator end-effector based on the manipulator base coordinate frame $base$ is obtained. The distance was selected as 100 mm to ensure sufficient margin for different size tomatoes. When the gripper is grasping, it must first reach point b which is 100 mm in front of the target tomato with the direction of the gripper opening towards tomato, and then it must reach the final harvesting point with the gripper opening towards the tomato to avoid the gripper closing to the tomato from other directions and collide with the tomato. The interpolation b in the manipulator end-effector coordinate frame is 100 mm in front of the tomato position a and the direction is the pose p of the manipulator end-effector. Let the vector from interpolation point b to the origin of $base$ coordinate frame as \mathbf{b} and k as the scaling factor. \mathbf{b} is obtained as

$$(1 - \frac{1}{k})|\mathbf{a}| = 0.1, \quad (44)$$

$$\mathbf{b} = \frac{1}{k}\mathbf{a} = \left[x_a \frac{|a| - 0.1}{|a|}, x_b \frac{|a| - 0.1}{|a|}, x_c \frac{|a| - 0.1}{|a|} \right]. \quad (45)$$

After vector \mathbf{b} is obtained, the coordinate of the interpolation point b can be obtained as $(x_a \frac{|a| - 0.1}{|a|}, x_b \frac{|a| - 0.1}{|a|}, x_c \frac{|a| - 0.1}{|a|})$. Subsequently, through the TF conversion relationship from

manipulator end-effector coordinate frame *tool* to the manipulator base coordinate frame *base*, the coordinate of interpolation point *b* under the base of the manipulator can be obtained. Interpolation point *b* is added in the trajectory planning of manipulator to ensure that the gripper is grasped from the gripper opening direction.

There are errors between the real position of tomato and the position calculated by acquisition owing to the camera and calibration errors. The errors may result in a collision between the end-effector of manipulator and the target tomato. If the harvesting robot approaches the tomato with a high speed, the tomato may be hit or damaged. If the overall grasping speed is low, it will result in a low grasping efficiency. In industrial applications, the quintic interpolation polynomial has not been used widely because it has no uniform speed section. However, the application of a quintic interpolation polynomial for end-effector speed planning on harvesting robot can adequately solve the problem that the end-effector of the manipulator contacting the tomato with high speed results in tomato damage.

The quintic interpolation polynomial requires six initial constraints. The initial desired position x_0 of the manipulator at t_0 and the target position x_n at t_n , the velocities of the initial position and target position v_0 and v_n , and the accelerations a_0 and a_n of the initial position and target positions jointly constitute the six basic constraints for the end trajectory planning of the manipulator.

$$\begin{cases} x(0) = x_0 \\ x(t_n) = x_n \end{cases}, \begin{cases} x'(0) = v_0 \\ x'(t_n) = v_n \end{cases}, \begin{cases} x''(0) = a_0 \\ x''(t_n) = a_n \end{cases} \quad (46)$$

To avoid the collision damage to the tomato caused by the end-effector of manipulator approach to the tomato with fast speed, the speed of the manipulator end-effector is limited when approaching the tomato. When the end-effector of manipulator approaches the tomato, the manipulator is controlled to reduce to a slow speed and a small acceleration. Speed and acceleration of the manipulator at the start and end positions is limited to 0, the velocity at the interpolation point *b* is limited to 20 mm/s, and the acceleration is limited to 0. The coefficients $c_0 - c_5$ can be obtained by substituting the above conditions into the quintic interpolation polynomial as

$$x(t) = c_0 + c_1t + c_2t^2 + c_3t^3 + c_4t^4 + c_5t^5. \quad (47)$$

The end-effector position, velocity, and acceleration at each time are obtained using quintic interpolation polynomials. The Jacobian matrix is converted to each joint of the manipulator to obtain the control quantity of each joint:

$$v = J(\theta)\theta', \quad (48)$$

where $v = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]$, $\theta' = [\theta'_1, \theta'_2, \theta'_3, \theta'_4, \theta'_5, \theta'_6]$. Similarly, the acceleration of each joint can be obtained.

Remark 2: The manipulator path is planned using interpolation point constraints to enable the manipulator to approach and grasp the tomato with the opening direction towards the tomato, avoiding the grasping failure caused by the gripper approaching the tomato from the side when the manipulator approaches the tomato. The end-effector speed of the manipulator is planned by quintic interpolation polynomial

programming to reduce the speed when the end-effector of manipulator is close to the tomato. The collision between the end-effector and the tomato with a high speed caused by position error is avoided.

G. Tomato grasping and separation

A soft gripper is used to grasp the tomato, and the end joint rotation of the manipulator is used to achieve the separation of tomato stems. Ripe tomato has a raised abscission layer at the junction of stalk and branch. The toughness of the abscission layer is significantly lower than the other stalks and branches. Rotating the ripe tomato by $90^\circ - 180^\circ$ can easily realize separation of tomatoes and stems without damaging tomatoes and branch (J. Liu et al., 2020). Therefore, the method of rotating the tomatoes is used to separate the stalks. Compared with the scissors-hand method of finding tomato stalks and then using scissors to cut stalks to separate them, the scissors-hands method requires a more accurate judgment on the position of stalks. It is often necessary to use point clouds and other methods for recognition and positioning, which significantly increases the amount of calculations. In addition, this method requires scissors to accurately reach the position of tomato stem for cutting, which also places high requirements on the control of manipulator. The gripper adopts a pneumatic soft structure, and the gripper adopts soft material to avoid damage to the tomato during the grasping. Compared with electric control, pneumatic control does not require an additional mechanism on the end-effector of manipulator, which can effectively reduce the mechanism and load of the manipulator's end-effector and make the grasping work more convenient.

To test performance of the proposed tomato recognition method based on background filtering, the recognition effect of the purely using HSV color segmentation algorithm and deep learning YOLOv4 algorithm is compared. Multiple experiments were performed under different experimental backgrounds and occlusion scenarios. Videos were captured using a RealSense D435i depth camera with 640×480 pixels. Experiments were performed using the harvesting robot equipped with an Intel Core i7-8850H 2.6 GHz CPU industrial computer.

In the absence of background interference, both color segmentation and deep learning algorithms could correctly recognize the target. When the background interference was large, as shown in Fig. 13(a), the tomato recognition algorithm based on color segmentation could not correctly recognize the tomato. The tomatoes or red area in the background interfered with recognition results in the wrong positioning of tomato center position. The larger the tomatoes and red range in the background, the greater the deviation is, which cause the harvesting robot to fail to grasp the tomato. Yolov4 can accurately recognize the tomato after background filtering.

As shown in Fig. 13(b) and Fig. 13(c), the background was filtered out and all pixels outside the manipulator workspace range were replaced with gray. The image after background filtering could only observe the objects in the manipulator workspace, which effectively eliminated background interference and reduced the complexity of the image. This method

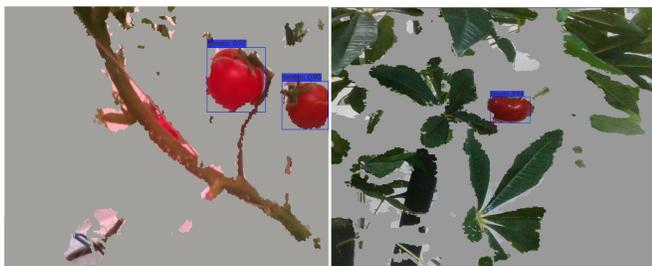
improved the recognition rate of tomato and effectively increased the recognition speed. 100 groups of experiments at different positions and angles by color segmentation were performed. The success rate of tomato recognition after filtering background was 96%, and the recognition speed was 20 frames/s.



(a) Recognition disturbed by background



(b) Tomato recognition after filtering out background interference



(c) YOLOv4 after filtering out background interference

Fig. 13. Tomato recognition based on background filtering.

H. Tomato grasping

UR3e and UR5 manipulator are used to harvest tomatoes in the non-dexterous workspace respectively to prove the applicability of the algorithm and present a more intuitive effect. The successful grasp using adaptive pose control method with the scene that cannot be reached using the fixed pose is compared. Let the manipulator reach the nearest point to the tomato with a fixed pose when the manipulator cannot reach with a fixed pose, Fig. 14 and Fig. 15 show the comparison effect. Different poses would lead to different harvesting results when harvesting tomatoes in a non-dexterous workspace. **Two different models of manipulators are used to verify the generalizability of our algorithm.**

Experimental data indicates that the adaptive end-effector pose control method can effectively improve the reachable range of the manipulator from the Fig. 14 and Fig. 15. The manipulator determines the end-effector pose that can

make it reach the target point that is in the non-dexterous workspace through the GA. The adaptive pose control method can enable the manipulator to grasp the tomatoes in the dexterous workspace, and significantly improve the success rate of tomato grasping in the range of 550-800 mm in a non-dexterous workspace. To further highlight the beneficial effect of the adaptive end-effector pose control method, the volume of the expanded grasping area with that of the original area is compared by

$$s = \frac{r_m^3 - r_p^3}{r_p^3}, \quad (49)$$

where r_m is the radius of the primary workspace, r_p is the radius of the dexterous workspace.

As shown in Fig. 16, the radius of UR3e manipulator grasping range was increased from 550 to 800 mm and UR5 manipulator grasping range was increased from 850 to 1050 mm. the grasping range of the UR3e manipulator was increased by 208% and the UR5 manipulator was increased by 89% through the calculation of (49), so as to improve the grasping success rate. The greater the ratio of the end-effector length to the radius of the dexterous workspace of the manipulator, the higher the ratio of the increase in the graspable range. The calculation time increased because determining the appropriate end-effector pose using the GA required a significant amount of time, but the calculation time can satisfy the requirements of the harvesting robot.

Tomato grasping experiments were performed to evaluate the proposed control method. 10 sets of experiments with a total of 400 grasps are conducted. Each set of experiments used the adaptive end-effector pose control method and the pose p for 20 grasps, and the position of each tomato was different. After each set of 40 grasps, the distance between the tomato and the manipulator was increased. The distance between the tomato and manipulator was the Euclidean distance from the position of the tomato center point to the origin of the manipulator base coordinate frame. The path planning algorithm used the RRT algorithm. The results are shown in Table II.

TABLE II
SUCCESS RATE OF MANIPULATOR REACHING TARGET POSITION BASED ON DIFFERENT POSE CONTROL METHODS

Number	Distance from target point to base(mm)	Success rate of fixed pose	Average time of fixed pose(ms)	Success rate of adaptive pose	Average time of adaptive pose(ms)
1	0-100	0	0	0	0
2	100-200	80%	5	100%	58
3	200-300	100%	4	100%	4
4	300-400	100%	4	100%	4
5	400-500	100%	4	100%	4
6	500-600	75%	4	100%	79
7	600-700	10%	5	100%	170
8	700-800	0	0	65%	1163
9	800-900	0	0	5%	4218
10	900-1000	0	0	0	0

In the experiment of group 1, the tomatoes were 0-100 mm away from the base of the manipulator. The manipulator



(a) Tomato grasping using the reachable grasping pose obtained by the adaptive pose algorithm

(b) Enlarged of (a)

(c) Failed grasping owing to unreachable poses

(d) Enlarged of (c)

Fig. 14. Comparison of reachable and unreachable grasping pose of UR3e manipulator.



(a) Tomato grasping using the reachable grasping pose obtained by the adaptive pose algorithm

(b) Enlarged of (a)

(c) Failed grasping owing to unreachable poses

(d) Enlarged of (c)

Fig. 15. Comparison of reachable and unreachable grasping pose of UR5 manipulator.

can not reach any center point of tomato because this range was within the singularity range of the manipulator. In the experiment of group 2, the tomatoes were 100-200 mm away from the base of the manipulator. The manipulator can reach most positions with the fixed pose method, and the success rate was 80% with an average time of 5 ms. The manipulator reached all center points of tomatoes with the adaptive end-effector pose, and the average time was 58 ms.

In the experiment of groups 3-5, the tomatoes were 200-500 mm away from the base of the manipulator. The manipulator can reach all center points of tomatoes with the fixed pose and adaptive end-effector pose, and the average time was 4 ms. The manipulator was far away from its own singular points and boundary range in this interval and can effectively reach

the target points.

In the experiment of group 6, the tomatoes were close to the boundary of the manipulator dexterous workspace, and the success rate of the fixed end-effector pose control method began to decline. The success rate was 75% when the tomatoes were 500-600 mm away from the base of the manipulator. The success rate decreased steeply to 10% in the experiment of group 7, in which the tomatoes were 600-700 mm away from the base of the manipulator. It was almost difficult to reach the target points, but the calculation time was 4 ms which was rapid. In the experiment of groups 6 and 7, the adaptive control method successfully reached all center points of tomatoes, but the calculation time increased rapidly, which were 79 and 170 ms.

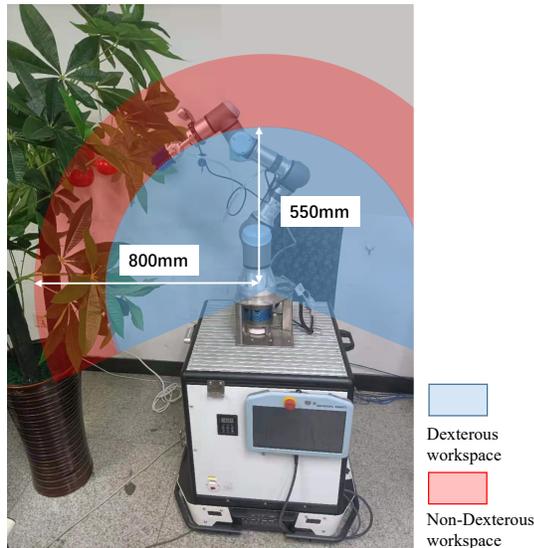


Fig. 16. Workspace of harvesting manipulator.

From the experiments of group 8, the fixed end-effector pose can not reach any target point. The success rate of the adaptive end-effector pose was 65% when the tomatoes were 700-800 mm away from the base of the manipulator, and the average calculation time was 1163 ms. In the experiment of group 9, the tomatoes were 800-900 mm away from the base of the manipulator. The success rate of the adaptive end-effector pose was 5% and the target point almost exceeded the workspace of the manipulator. In the experiment of group 10, the tomatoes position completely exceeded the workspace of the manipulator, and the target point can not be reached using any pose control method.

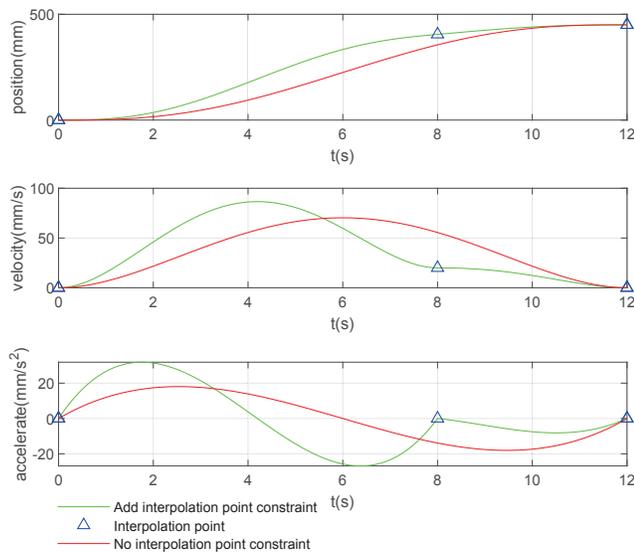


Fig. 17. Position, speed and acceleration of end-effector.

The relationship between the position, velocity, acceleration and time of the end-effector obtained by trajectory planning using the quintic interpolation polynomial is shown in Fig. 17. The green curve is the trajectory curve with velocity



(a) The gripper is close to the tomato (b) Failed to grasp due to collision from the side with tomato



(c) The opening of the clamping claw (d) Successfully grasp the tomato is toward the tomato

Fig. 18. The manipulator grasp the tomato successfully by add interpolation points.

constraint added at interpolation point b , and the red curve is the trajectory curve without the interpolation point constraint. The figure shows that the two curves reach the same position simultaneously, and the curves can remain smooth without a jump. The green curve with interpolation points reaches interpolation point b at a faster speed and decreases to a lower speed. Finally, it reaches the tomato position at a slower speed and an acceleration close to 0, which ensures the safe grasping of the tomato and avoids the damage of the tomato by the end-effector.

The method of tomato grasping by adding interpolation points was verified by experiments. As shown in Fig. 18 (a)-(b), the harvesting manipulator grasped the tomato from the side resulting in failed harvesting. As show in Fig.18 (c)-(d), the end-effector approached the tomato from the side, and the tomato was pushed aside by the end-effector without being wrapped by the end-effector, resulting in the failure of grasping.

As shown in Fig. 19, the adaptive end-effector pose control method and manipulator path control method with interpolation point constraints proposed in this paper were used for tomato harvesting. After the harvesting-robot recognizes the tomato, it calculates the appropriate grasping pose p , which is not a reachable pose. The reachable pose is obtained by using adaptive pose control algorithm and adds 0.1 m in front of the tomato in the pose as an interpolation point for trajectory planning. The manipulator approaches from the front of the tomato according to the planned trajectory and completed the grasp, places the grasped tomato into the harvest basket and finally returns to the initial pose to complete a round of grasping tasks and ready for the next tomato harvesting.

100 harvesting experiments on tomatoes in different posi-

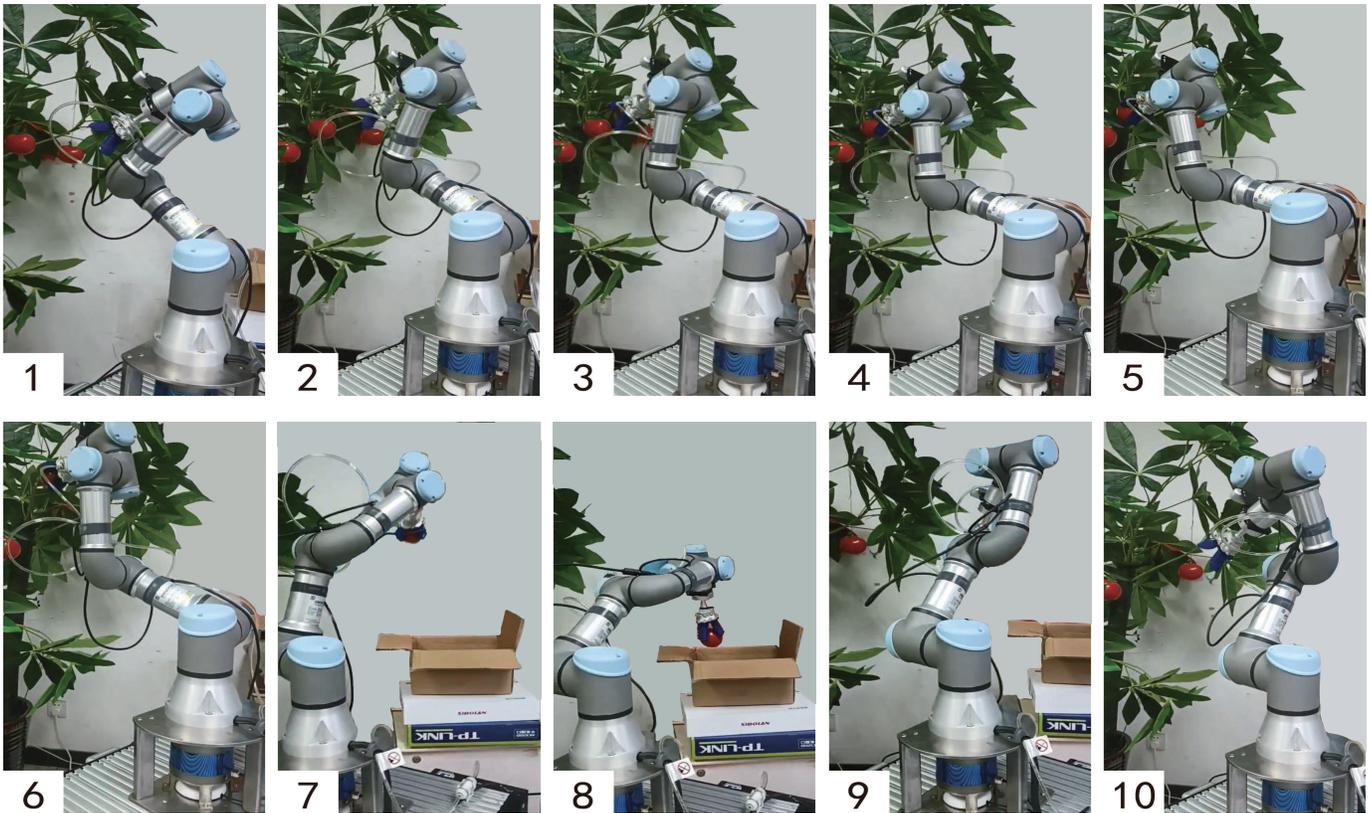


Fig. 19. Tomato harvesting with adaptive end-effector pose control method and path control method with interpolation points.

tions were performed, and 88 of these were successful. The main reason for the failure was that the vibration caused by the grasping induced other tomatoes positioning errors. The experimental results show that the robot harvesting success rate is 88%, and the average time for the robot to complete a cycle of grasping task is approximately 20 s.

IV. CONCLUSION

(1) The reachable end-effector pose of the harvesting manipulator is calculated using a GA to enable the manipulator's end-effector reach the tomato position when the tomatoes are in a non-dexterous workspace such as the boundary of the manipulator. The radius of the workspace of the UR3e manipulator with the end-effector increased to 550-800 mm and the workspace of the UR3e manipulator with end-effector installed expands by 208%. Harvesting robot completes a cycle of grasping task is 20 s.

(2) The quintic interpolation polynomial is used for the trajectory planning of the end-effector. The interpolation point is added to constrain the pose and speed of the end-effector at the interpolation point to enable the end-effector to slowly approach the tomato in the direction of opening towards the tomato, which reduces the tomato damage rate.

(3) Tomato-harvesting experiments were performed using the three methods proposed in this paper. The results show that the tomato recognition is not disturbed by the background, the grasping range is larger, the end-effector no longer collides with the tomato, and the tomato grasping can be performed

successfully. The experiments verified the effectiveness and feasibility of the proposed algorithms.

V. FUTURE WORK

(1) The largest failure in the experiment was due to the shaking of tomatoes caused by grasping, resulting in the continuous shaking of the position of the remaining tomatoes in the process of grasping, which finally resulted in the failure of grasping. Therefore, the prediction algorithm of shaking tomato will be designed to improve the success rate of grasping shaking tomato in the future.

(2) The selection of tomato for harvesting in the same area is random, resulting in low harvesting efficiency currently. In the future, a method of planning a tomato-harvesting sequence will be designed to optimize the tomato-harvesting sequence and improve the harvesting efficiency in combination with the information of tomato distribution in the area.

REFERENCES

- Arad, B., Balendonck, J., Barth, R., Ben-Shahar, O., Edan, Y., Hellström, T., ... others (2020). Development of a sweet pepper harvesting robot. *Journal of Field Robotics*, 37(6), 1027–1039.
- Aristidou, A., & Lasenby, J. (2011). Fabrik: A fast, iterative solver for the inverse kinematics problem. *Graphical Models*, 73(5), 243–260.

- Bac, C. W., van Henten, E. J., Hemming, J., & Edan, Y. (2014). Harvesting robots for high-value crops: State-of-the-art review and challenges ahead. *Journal of Field Robotics*, 31(6), 888–911.
- Barth, R., Hemming, J., & Van Henten, E. J. (2019). Angle estimation between plant parts for grasp optimisation in harvest robots. *Biosystems Engineering*, 183, 26–46.
- Beeson, P., & Ames, B. (2015). Trac-ik: An open-source library for improved solving of generic inverse kinematics. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)* (pp. 928–935).
- Benavides, M., Cantón-Garbín, M., Sánchez-Molina, J., & Rodríguez, F. (2020). Automatic tomato and peduncle location system based on computer vision for use in robotized harvesting. *Applied Sciences*, 10(17), 5887.
- Callejón-Ferre, Á. J., Montoya-García, M. E., Pérez-Alonso, J., & Rojas-Sola, J. I. (2015). The psychosocial risks of farm workers in south-east Spain. *Safety Science*, 78, 77–90.
- Cao, B., Sun, K., Gu, Y., Jin, M., & Liu, H. (2018). Workspace analysis based on manipulator pose dexterity map. In *2018 3rd International Conference on Robotics and Automation Engineering (ICRAE)* (pp. 166–170).
- Diankov, R. (2010). Automated construction of robotic manipulation programs.
- Feng, Q., Wang, X., Wang, G., & Li, Z. (2015). Design and test of tomatoes harvesting robot. In *2015 IEEE International Conference on Information and Automation* (pp. 949–952).
- Fujinaga, T., Yasukawa, S., & Ishii, K. (2019). System development of tomato harvesting robot based on modular design. *IEICE Proceedings Series*, 57(SS1-1).
- Gongal, A., Amatya, S., Karkee, M., Zhang, Q., & Lewis, K. (2015). Sensors and systems for fruit detection and localization: A review. *Computers and Electronics in Agriculture*, 116, 8–19.
- Hayashi, S., Shigematsu, K., Yamamoto, S., Kobayashi, K., Kohno, Y., Kamata, J., & Kurita, M. (2010). Evaluation of a strawberry-harvesting robot in a field test. *Biosystems Engineering*, 105(2), 160–171.
- Ishii, K., Matsuo, T., Takemura, Y., Sonoda, T., Nishida, Y., Yasukawa, S., & Fujinaga, T. (2021). Tomato-harvesting-robot competition towards smart agriculture. In *Proceedings of International Conference on Artificial Life & Robotics (ICAROB2021)* (pp. 1–5).
- Lehnert, C., McCool, C., Sa, I., & Perez, T. (2020). Performance improvements of a sweet pepper harvesting robot in protected cropping environments. *Journal of Field Robotics*, 37(7), 1197–1223.
- Lili, W., Bo, Z., Jinwei, F., Xiaoan, H., Shu, W., Yashuo, L., ... Chongfeng, W. (2017). Development of a tomato harvesting robot used in greenhouse. *International Journal of Agricultural and Biological Engineering*, 10(4), 140–149.
- Lin, G., Tang, Y., Zou, X., Cheng, J., & Xiong, J. (2020). Fruit detection in natural environment using partial shape matching and probabilistic Hough transform. *Precision Agriculture*, 21(1), 160–177.
- Ling, X., Zhao, Y., Gong, L., Liu, C., & Wang, T. (2019). Dual-arm cooperation and implementing for robotic harvesting tomato using binocular vision. *Robotics and Autonomous Systems*, 114, 134–143.
- Liu, G., Nouaze, J. C., Touko Mbouembe, P. L., & Kim, J. H. (2020). Yolo-tomato: A robust algorithm for tomato detection based on yolov3. *Sensors*, 20(7), 2145.
- Liu, J., Peng, Y., & Faheem, M. (2020). Experimental and theoretical analysis of fruit plucking patterns for robotic tomato harvesting. *Computers and Electronics in Agriculture*, 173, 105330.
- McCool, C., Sa, I., Dayoub, F., Lehnert, C., Perez, T., & Upcroft, B. (2016). Visual detection of occluded crop: For automated harvesting. In *2016 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 2506–2512).
- Qiu, S., Wang, M., & Kermani, M. R. (2021). A fast and accurate new algorithm for hand-eye calibration on $so(3) \times r^3$. *Control Engineering Practice*, 109, 104726.
- Rauchfuss, J. W., & Yang, D. C. (2000). A geometric approach for determining exact point accessibility of robotic manipulations. *J. Mech. Des.*, 122(3), 287–293.
- Siciliano, B., & Khatib, O. (2016). *Springer handbook of robotics*. Springer.
- Silwal, A., Davidson, J. R., Karkee, M., Mo, C., Zhang, Q., & Lewis, K. (2017). Design, integration, and field evaluation of a robotic apple harvester. *Journal of Field Robotics*, 34(6), 1140–1159.
- Universal ROBOTS. (2022). *Ur3e, A NEW SERIES OF COLLABORATIVE ROBOTS*. (<http://www.universal-robots.com/products/ur3e-robot/>), Last accessed on 2022-07-25)
- Wang, Y., Yang, Y., Yang, C., Zhao, H., Chen, G., Zhang, Z., ... Xu, H. (2019). End-effector with a bite mode for harvesting citrus fruit in random stalk orientation environment. *Computers and Electronics in Agriculture*, 157, 454–470.
- Yaguchi, H., Nagahama, K., Hasegawa, T., & Inaba, M. (2016). Development of an autonomous tomato harvesting robot with rotational plucking gripper. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 652–657).
- Ye, L., Duan, J., Yang, Z., Zou, X., Chen, M., & Zhang, S. (2021). Collision-free motion planning for the litchi-picking robot. *Computers and Electronics in Agriculture*, 185, 106151.
- Zhao, Y., Gong, L., Huang, Y., & Liu, C. (2016). A review of key techniques of vision-based control for harvesting robot. *Computers and Electronics in Agriculture*, 127, 311–323.