Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 3105909, 12 pages
https://doi.org/10.1155/2022/3105909

Research Article

Artificial Intelligence in Agricultural Picking Robot Displacement Trajectory Tracking Control Algorithm

Zhipan Wu[1] **and Huaying Du**[2]

[1]*School of Computer Science and Engineering, Huizhou University, Huizhou, 516007 Guangdong, China*
[2]*School of Information Technology, City College of Huizhou, Huizhou, 516025 Guangdong, China*

Correspondence should be addressed to Huaying Du; duhuaying@tm.hzc.edu.cn

Received 26 April 2022; Revised 23 May 2022; Accepted 2 June 2022; Published 16 June 2022

Academic Editor: Chia-Huei Wu

With the development and leap of artificial intelligence technology, more and more robots have penetrated into all walks of life. Today's agriculture is changing in the direction of modernization and automation. On the one hand, because of rising labor costs, it cannot afford to consume a large amount of labor for agricultural operations. On the other hand, the population is growing rapidly, and traditional agricultural production, picking, and other links have been unable to keep pace with the development of the times. Therefore, it is very necessary to use artificial intelligence technology to transform traditional agriculture. The purpose of this paper is to use artificial intelligence technology to plan, track, and optimize the displacement trajectory of the agricultural picking robot, so as to improve the working efficiency of the picking robot. In this paper, the neural network, the D-H modeling method of the manipulator, and the forward and reverse motion of the manipulator are explained and analyzed, and based on the relevant algorithms of neural network, the manipulator is modeled, and then the, forward and reverse motion of the manipulator is analyzed in detail, and the digital model of the picking robot is constructed. Then, the angle and motion speed of each joint of the robot are analyzed to reduce the motion trajectory error caused by friction and other factors. Then, the simulation experiment of the displacement trajectory tracking control is carried out, and the linear trajectory motion and the arc trajectory motion are deeply analyzed, and the axis error is greatly reduced after 6 iterations. Finally, the displacement trajectory is optimized. The optimized total movement time is shortened by 6.84 seconds, which enables the picking robot to not only ensure work efficiency but also accurately complete the planned displacement trajectory. After continuous experiments on the algorithm model and the picking robot, the actual trajectory of the picking robot at 0.7 seconds can be expected. The trajectories are completely coincident, indicating that the neural network plays a very important role in the trajectory research of picking robots.

1. Introduction

In recent years, the population has grown substantially, and traditional agricultural production has been unable to meet the growing demand for agricultural products. At present, the agricultural production, picking, and transportation technologies in many areas are relatively outdated. It requires a lot of manpower and material resources, which is time-consuming and labor-intensive. At the same time, it is accompanied by the pressure of agricultural resources and the shortage of labor resources. Modern agriculture urgently needs new technologies to transform and upgrade. At this time, artificial intelligence technology can play a great role in the agricultural field. Picking robots are playing an important role in intelligent agriculture. Picking robots can be programmed to achieve tasks such as picking, handling, and boxing of fruits and vegetables. It not only recognizes and locates crops quickly but also has high work efficiency and can complete high-intensity picking tasks. It plays a very important role in saving labor costs, reducing labor risks, realizing agricultural modernization, and increasing the total agricultural output value. At the same time, the development of picking robots provides theoretical support and practical experience for the improvement and

application of artificial intelligence technology. Therefore, it is the general trend to promote the development of agriculture to study the displacement trajectory tracking control technology of agricultural picking robots.

Agricultural development is moving towards large-scale and intelligent modernization. The cost of agricultural labor is gradually rising, which will inevitably lead to a shortage of labor. Therefore, it is essential for artificial intelligence to enter the production and transportation links of agriculture. The planting, picking, transportation, and processing of crops are becoming automated and mechanized. Traditional agricultural picking is labor-intensive, expensive, and inefficient. The modern agricultural picking robot can greatly reduce the labor intensity of farmers, save costs, improve picking efficiency, ensure that fruits and vegetables are harvested within an appropriate time, and avoid spoilage and waste caused by too late picking. The development of picking robots not only adds power to agricultural production but also greatly promotes the reform and upgrading of agricultural management models. In the past, all links required labor, and now the whole machine can be operated, and labor costs have been greatly reduced. At the same time, the research on picking robot science and technology also provides practical experience for the improvement and development of artificial intelligence. This has far-reaching significance for improving the growth of agricultural output value and the development of artificial intelligence.

The innovations of this paper are as follows: (1) It uses artificial intelligence technology to track the displacement trajectory of the agricultural picking robot, control the moving direction and speed of the picking robot, and improve the picking efficiency. (2) The change matrix is used to describe the pose, which simplifies the process of robot displacement planning. (3) Combined with the theory of iterative operation, the axis error is used to mediate and improve the tracking of the displacement trajectory. The trajectory error obtained by the experiment is processed and transmitted to the controller, and the controller operates the robot to repeat the experimental operation continuously, and the error is reduced through continuous experiments, and the accuracy of the tracking control of the displacement trajectory is improved.

2. Related Work

Many scholars have paid attention to the research on the control algorithm of robot displacement trajectory tracking. Ou et al. discussed the fixed-time tracking control of nonholonomic wheeled mobile robots based on visual servoing. They proposed a tracking error system between the mobile robot and the desired trajectory, enabling the robot to track the reference trajectory in a fixed time. However, this process is complicated and may result in inaccurate results [1]. Ma et al. introduced the U-model method to relieve the requirement of dynamic mathematical model and simplify the design of the trajectory tracking controller of the manipulator. However, there are certain errors in this method, resulting in insufficient accuracy [2]. Peng et al. proposed a controlled DAE-based optimal control (IOC) method. A series of IOC problems tend to be continuous trajectory tracking problems at each time step, and a Linear Complementarity Problem (LCP) is derived to solve the IOC problem. This method provides a unified framework for solving the trajectory tracking control problem of robotic multibody dynamic systems. However, this method is relatively old, the data obtained is slightly lacking, and there is a lack of innovation [3]. Khalilpour et al. proposed a novel feedback method. On the premise of dynamic visual recognition, the joint motion sensor is optimized to improve the performance of trajectory tracking, and the oscillation is reduced through the operation of the end effector. The authors derive kinetic formulations for large-scale deployable cable-driven robots. The model can better satisfy the stability condition [4]. Wang et al. proposed a wavelet neural network fuzzy sliding mode controller for the nonlinear problem of the manipulator system. The experimental results show that the proposed control scheme is suitable for 3-DOF robot displacement trajectory tracking control. But this method has higher requirements on the accuracy of the algorithm model [5]. Bensafia et al. utilize Fractional Model Reference Adaptive Controller (FMRAC) to achieve smooth trajectories and optimize the performance of the robotic arm of the SCARA robot. However, this operation has requirements on the data analysis environment and needs to be performed in a stable environment [6]. He et al. optimize the online computation of MPC using a model predictive control- (MPC-) based tracking controller. The method includes threshold curve-based event triggering and threshold band-based event triggering. However, this method is based on a large amount of historical data and requires an accurate database as support [7].

3. Picking Robot Trajectory Tracking Control Algorithm

3.1. Radial Basis Neural Network (RBF). Radial basis neural network is a feed-forward neural network model without feedback. It is usually presented as a three-layer feedforward network: input layer, hidden layer, and output layer [8]. Input nodes can transmit a large amount of information from the outside world to the neural network. The hidden layer maps these input variables into a nonlinear space [9]. In the hidden layer space, the input variables will then be transmitted, and the activation function will be generated after the action. This activation function is the radial basis function, which corresponds to the hidden layer neurons one-to-one [10, 11]. If the number of hidden layer nodes in the radial basis neural network increases, the solving ability and approximation ability of the corresponding network will be stronger, and the structure will be more complex. The structure of the output layer is relatively simple. The linear weighted sum of the output of the hidden nodes obtained after continuous calculation is the output result [12]. Figure 1 is the basic structure of radial basis neural network.

Supposing j represents the number of input nodes, n represents the number of hidden layer nodes, and k represents the number of output nodes. The input vector is denoted as $A = [a_1, a_2, \cdots a_j]^P$, the output matrix is denoted as $\lambda = [\lambda(a, b_1), \lambda(a, b_2), \cdots, \lambda(a, b_n)]$, b_i is the node position, and the ith hidden layer, $i = 1, 2, \cdots, n$, is presented. The output $Q_{nk} =$

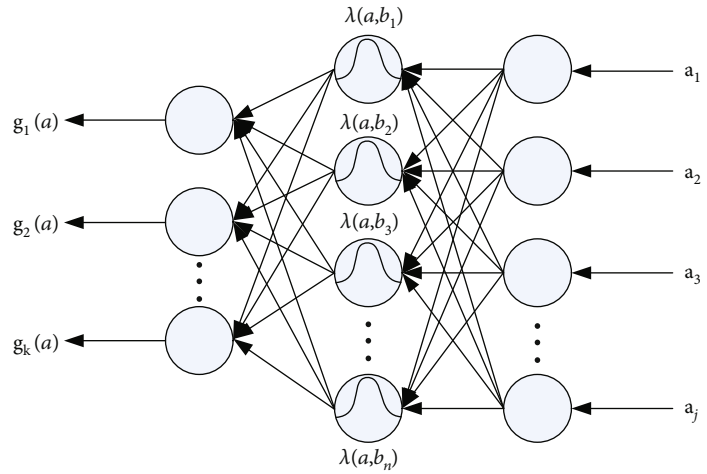

Figure 1: Radial basis neural network structure.

$[Q_1, Q_2, \cdots, Q_n]^P$ is the weight matrix, where $Q_i = [Q_{i1}, Q_{i2}, \cdots, Q_{ik}]^P$, and the vector is $G(A) = [g_1(a), g_2(a), \cdots, g_k(a)]^p$.

There are the following three expressions of the activation function:

(1) Multiple quadratic functions

$$\lambda(a, b_l) = \left(a^2 + b_l^2\right)^{12}, \quad l = 1, 2, \cdots, n \tag{1}$$

(2) Inverse multiple quadratic function

$$\lambda(a, b_l) = \frac{1}{\left(a^2 + b_l^2\right)^{12}}, \quad l = 1, 2, \cdots, n \tag{2}$$

(3) Gaussian function

$$\lambda(a, b_l) = \exp\left(-\frac{a - b_l^2}{\mu^2}\right), \quad l = 1, 2, \cdots, n \tag{3}$$

The input value is set to A, the center of the basis function is set to b_l, and the lth is presented. The smoothing factor is μ. l is the number of hidden layer centers, and $a - b_l$ is the norm. $\lambda(a, b_l)$ has a maximum value in b_l, and there is one and only this one. If the value of $a - b_l$ increases, $\lambda(a, b_l)$ becomes zero.

The output layer is calculated by the hidden layer:

$$g_s(a) = \sum_{l=1}^{n} Q_{ls}\lambda(a, b_l), \quad S = 1, 2, \cdots, k. \tag{4}$$

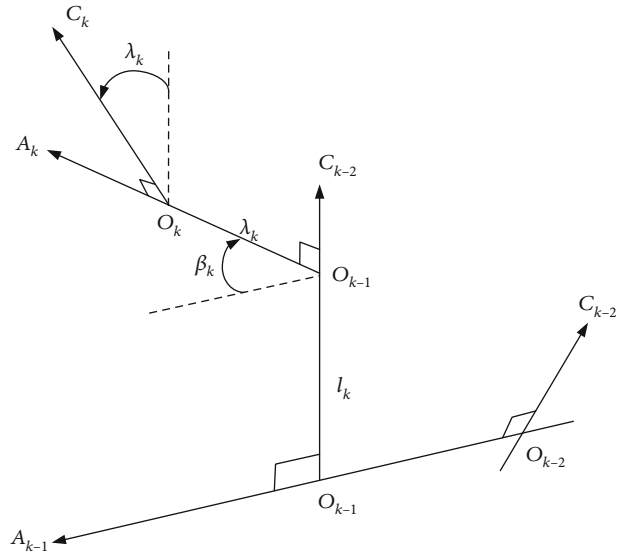

Figure 2: Schematic diagram of rod parameters.

Among them, Q_{ls} is represented as a weight value, which is between the lth hidden layer node and the Sth output layer node.

As the variables of the input layer of the Gaussian function increase, the system does not become more complex. And the smoothness of the Gaussian function is good, which is convenient for theoretical analysis [13]. Therefore, the Gaussian function is a calculation function often used by the radial basis function.

3.2. D-H Modeling Method of Robotic Arm

3.2.1. Parameters. The robotic arm D-H establishes a coordinate system on each link. The relationship between adjacent coordinate systems is represented by a matrix transformation [14–17]. After transformation, the pose of the end effector is obtained, and the pose is represented by base coordinates.

The adjacent coordinate systems and links are represented by the second transformation matrix. If the motion analysis of

the manipulator is to be performed, each link must be analyzed and described [18–20]. r_k is the rod length and λ_k is the torsion angle. In addition, l_k is the offset distance, indicating the distance between the two links, and β_k is the joint angle, indicating the number of angles formed by the normal lines of the two links. The resulting parameters are shown in Figure 2.

3.2.2. Establish a Coordinate Axis. The pose relationship between each link is a coordinate system fixed on each link. The coordinate system $\{P\}$ is the coordinate system of the end of the manipulator, and $\{k-1\}$ is the coordinate system fixed to the link k.

For a rotary joint, the joint parameters are l_k, r_k, and λ_k, and the joint variable is β_k. For moving joints, the joint parameters are β_k, r_k, and λ_k, and the joint variable is l_k.

From this, the relationship between the coordinate system $\{k\}$ and the coordinate system $\{k-1\}$ is

$$
\begin{aligned}
{}^{k-1}G_k &= \mathrm{Rot}(B,\beta_k)\mathrm{Trans}(0,0,l_k)\mathrm{Trans}(c_k,0,0)\mathrm{Rot}(A,\lambda_k) \\
&= \begin{bmatrix} \mathrm{d}\beta_k & -\mathrm{d}\lambda_k \mathrm{e}\beta_k & \mathrm{e}\lambda_k e\beta_k & \lambda_k \mathrm{d}\beta_k \\ \mathrm{e}\beta_k & \mathrm{d}\lambda_k \mathrm{d}\beta_k & -\mathrm{e}\lambda_k \mathrm{d}\beta_k & \lambda_k \mathrm{e}\beta_k \\ 0 & \mathrm{e}\lambda_k & \mathrm{d}\lambda_k & \mathrm{l}_k \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned} \tag{5}
$$

${}^{k-1}G_k$ is a homogeneous transformation matrix between two adjacent coordinate systems $\{k-1\}$ and coordinate system $\{k\}$.

The homogeneous transformation matrix ${}^0U_{\mathrm{k}}$ of any coordinate system $\{k\}$ relative to the base coordinate system is the continuous product of each adjacent homogeneous transformation matrix ${}^0U_{\mathrm{k}}$, written as

$$
\begin{aligned}
{}^0U_k &= {}^0G_1\,{}^0G_2 \cdots {}^{k-1}G_k = \prod_{i-1}^{k} {}^{i-1}G_i \\
&= \begin{bmatrix} \mathrm{p}_k & \mathrm{t}_k & \mathrm{r}_k & \mathrm{v}_k \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^0F_k & {}^0\mathrm{v}_k \\ 0 & 1 \end{bmatrix}.
\end{aligned} \tag{6}
$$

The attitude matrix of the kth coordinate system is $[p_k t_k \mathrm{r}_k]$, and the position vector of the kth coordinate system is v_k.

3.2.3. Establish the Motion Equation of the Manipulator. A five-degree-of-freedom serial robot is established, and the five joints are rotatable, as shown in Figure 3.

Synthetically, the coordinate system is established, the base coordinate system {0} is taken, and the hand coordinate system is {5}, and Figure 4 is obtained.

According to the coordinate system, D-H is established as shown in Table 1.

It can be seen from Table 1 that among the five joints, the fifth robotic arm used for picking operations has the longest offset distance, and the second and third joints used to adjust the picking distance have the longest rod lengths. The first and fourth joints can be used to adjust the angle of picking.

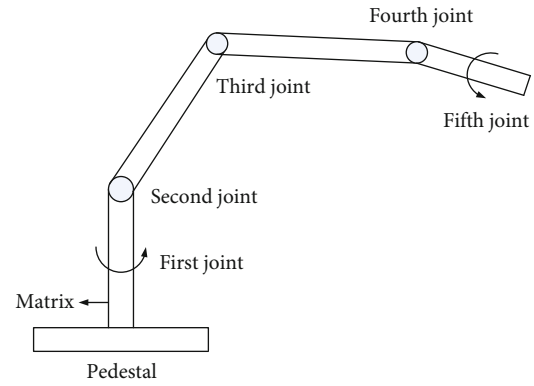

Figure 3: Structure diagram of the robotic arm.

3.3. Picking Robot Solution

3.3.1. The Robotic Arm Is Moving. The forward motion of the manipulator is the pose of the manipulator end effector in the coordinate system [21].

The total transformation matrix from the base coordinate system to the hand coordinate system is

$$
{}^0U_k = {}^0G_1^1 G_2^2 G_3^3 G_4^4 G_5 = \begin{bmatrix} \mathrm{q}_a & \mathrm{o}_a & \mathrm{r}_a & \mathrm{v}_a \\ \mathrm{q}_\mathrm{b} & \mathrm{o}_\mathrm{b} & \mathrm{r}_b & \mathrm{v}_b \\ \mathrm{q}_c & \mathrm{o}_c & \mathrm{r}_c & \mathrm{v}_c \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{7}
$$

Among them,

$$
\begin{aligned}
q_a &= e\beta_1 e\beta_5 + d\beta_1 d\beta_{234} d\beta_5, \\
q_b &= e\beta_1 \mathrm{d}\beta_{234} d\beta_5 - d\beta_1 e\beta_5, \\
q_c &= -e\beta_{234} d\beta_5, \\
o_a &= \mathrm{e}\beta_1 \mathrm{d}\beta_5 - \mathrm{d}\beta_1 \mathrm{d}\beta_{234} \mathrm{e}\beta_5, \\
o_b &= -d\beta_1 d\beta_5 - e\beta_1 d\beta_{234} e\beta_5, \\
o_c &= e\beta_{234} e\beta_5, \\
r_a &= -d\beta_1 e\beta_{234}, \\
r_b &= -e\beta_1 e\beta_{234}, \\
\mathrm{r}_c &= -d\beta_{234}, \\
v_a &= r_2 d\beta_1 d\beta_2 - l_5 d\beta_1 e\beta_{234} - r_3 d\beta_1 d\beta_{23}, \\
v_b &= r_2 e\beta_1 d\beta_2 - l_5 e\beta_1 e\beta_{234} - r_3 e\beta_1 d\beta_{23}, \\
v_c &= -r_2 e\beta_2 - l_5 d\beta_{234} - r_3 e\beta_{23}.
\end{aligned} \tag{8}
$$

$\beta_{23} = \beta_2 + \beta_3$, $\beta_{234} = \beta_2 + \beta_3 + \beta_4$, and so on.

From this, the pose of the hand coordinates in the base coordinate system can be obtained.

3.3.2. Reverse Motion of the Robotic Arm. The inverse motion of the manipulator is the joint variable obtained based on the known pose of the manipulator end in the coordinate system [22]. Using the algebraic method, the

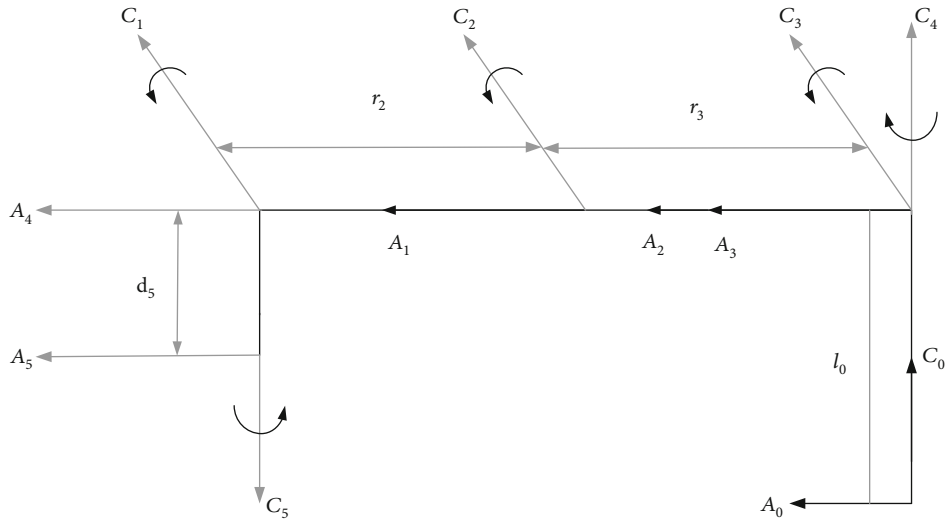

Figure 4: D-H method coordinate system.

Table 1: D-H data table of each joint.

Joint	Joint angle β_k	Offset l_k	Rod length r_k	Torsion angle λ_k
#1	β_1	0	0	-90
#2	β_2	0	393	0
#3	β_3	0	361	0
#4	β_4	0	0	-90
#5	β_5	274	0	0

joint variables β_1, β_2, β_3, β_4, and β_5 of the inverse motion are obtained.

Multiplying formula (7) by the left of ${}^0G_1^{-1}$ yields

$$ {}^0G_1{}^{-1}U = {}^1G_2{}^2G_3{}^3G_4{}^4G_5 = {}^1U_5. \tag{9} $$

Both ends are represented as

$$ \begin{bmatrix} \mathrm{d}\beta_1 & e\beta_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -\mathrm{e}\beta_1 & \mathrm{d}\beta_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_\mathrm{a} & o_\mathrm{a} & r_\mathrm{a} & v_\mathrm{a} \\ q_\mathrm{b} & o_\mathrm{b} & r_\mathrm{b} & v_\mathrm{b} \\ q_\mathrm{c} & o_\mathrm{c} & r_\mathrm{c} & v_\mathrm{c} \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^1U_5. \tag{10} $$

Putting (3,4) on both sides of formula (10), it gets

$$ d\beta_1 \mathrm{v}_c - e\beta_1 \mathrm{v}_a = 0. \tag{11} $$

With trigonometric transformation, it gets

$$ v_a = \mu \cos\omega, v_a = \mu \sin\omega. \tag{12} $$

Among them, $\mu = \sqrt{v_a^2 + v_b^2}, \omega = r \tan 2(v_b, v_a)$.

It can be derived as follows:

$$ \beta_1 = r \tan 2(v_b, v_a). \tag{13} $$

Formula (7) both sides (1,3) and (2,3) get

$$ \frac{-e\beta_{234}}{d\beta_{234}} = \frac{r_a d\beta_1 + r_b e\beta_1}{-r_c}. \tag{14} $$

By solving, it can get

$$ \beta_{234} = \mathrm{rtan2}(\mathrm{r_a d}\beta_1 + \mathrm{r_b e}\beta_1, \mathrm{r_c}). \tag{15} $$

Formula (7) on both sides (1,4) and (2,4), it can get

$$ \begin{cases} r_2 d\beta_2 + r_3 d\beta_{23} = d\beta_1 v_a + e\beta_1 v_b + l_5 e\beta_{234}, \\ r_2 e\beta_2 + r_3 e\beta_{23} = -v_c - l_5 d\beta_{234}. \end{cases} \tag{16} $$

Letting $d\beta_1 v_a + e\beta_1 v_v + l_5 e\beta_{234} = M$ and $-v_c - l_5 d\beta_{234} = N$, get

$$ c\beta_3 = \frac{M^2 + N^2 - r_2^2 - r_3^3}{2r_2 r_3}. \tag{17} $$

Letting $d\beta_3 = W$, then

$$ e\beta_3 = \pm\sqrt{1 - W^2}, \quad \beta_3 = r \tan 2\left(\pm\sqrt{1 - W^2}, W\right). \tag{18} $$

β_3 has two solutions, which are represented by positive and negative signs.

Left-multiplying ${}^0G_1^{-1}$ by formula (7), we get

$$ {}^0G_1{}^{-1}U = {}^3G_4{}^4G_5 = {}^3U_5. \tag{19} $$

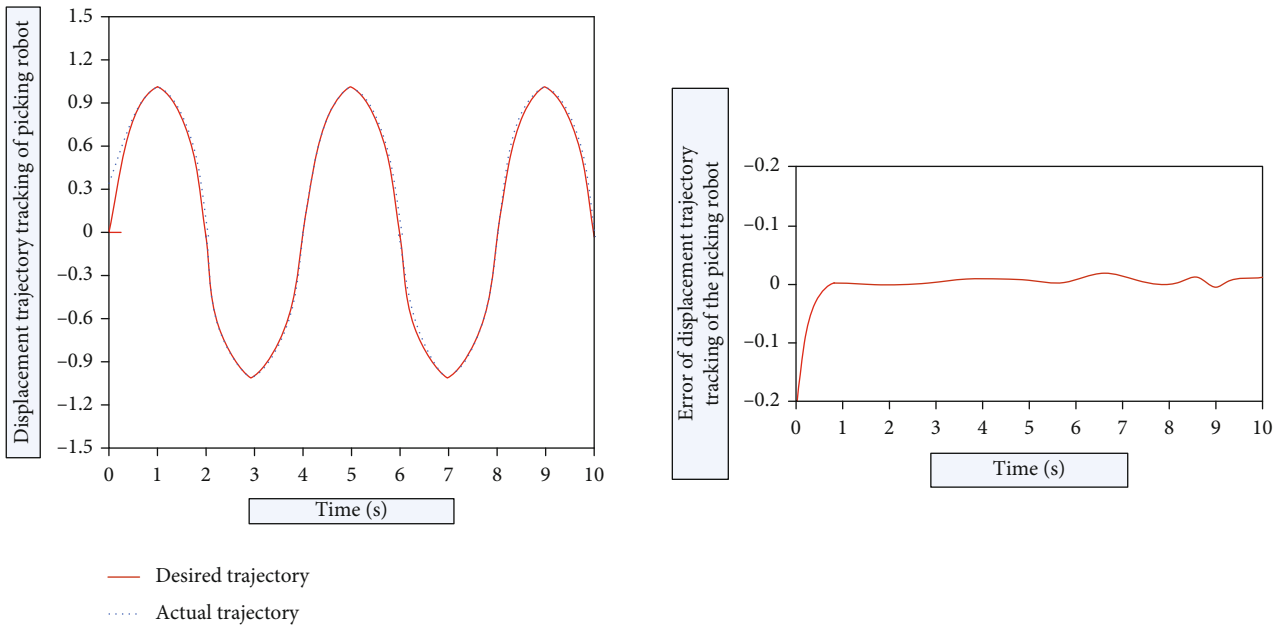

Figure 5: Displacement trajectory tracking and tracking error of picking robot.

Both ends get

$$\begin{bmatrix} d\beta_1 d\beta_{23} & e\beta_1 d\beta_{23} & -e\beta_{23} & -(r_3 + r_2 d\beta_3) \\ -d\beta_1 e\beta_{23} & -e\beta_1 e\beta_{23} & -d\beta_{23} & r_2 e\beta_3 \\ -e\beta_1 & d\beta_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_a & o_a & r_a & v_a \\ q_b & o_b & r_b & v_b \\ q_c & o_c & r_c & v_c \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^3U_5. \tag{20}$$

Equating both ends (1,3) and (2,3) of formula (20), it gets

$$\begin{cases} e\beta_4 = e\beta_2(r_a d\beta_1 e\beta_3 + r_c d\beta_3 + r_b e\beta_1 e\beta_3) - d_2(r_a d\beta_1 d\beta_3 + r_c e\beta_3 + r_b e\beta_1 d\beta_3), \\ d\beta_4 = -e\beta_2(-r_a e\beta_3 + r_a d\beta_1 d\beta_3 + r_b e\beta_1 d\beta_3) + d_2(r_a d\beta_3 + r_a d\beta_1 e\beta_3 + r_b e\beta_1 e\beta_3). \end{cases} \tag{21}$$

Letting $r_c = H$ and $r_a d\beta_1 + r_b e\beta_1 = F$ be substituted into formula (21) to get

$$\begin{cases} e\beta_4 = e\beta_2(Fe\beta_3 + He\beta_3) - d\beta_2(Fd\beta_3 - He\beta_3), \\ d\beta_4 = e\beta_2(He\beta_3 - Fd\beta_3) - d_2(Hd\beta_3 + Fe\beta_3). \end{cases} \tag{22}$$

Multiplying ${}^0G_1^{-1}$ to the left by formula (7), it gets

$${}^0G_1{}^{-1}U = {}^2G_3{}^3G_4{}^4G_5 = {}^2U_5. \tag{23}$$

Both ends are obtained as

$$\begin{bmatrix} d\beta_1 d\beta_2 & e\beta_1 d\beta_2 & -e\beta_2 & -r_2 \\ -d\beta_1 e\beta_2 & -e\beta_1 e\beta_2 & -d\beta_{23} & 0 \\ -e\beta_1 & d\beta_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_a & o_a & r_a & v_a \\ q_b & o_b & r_b & v_b \\ q_c & o_c & r_c & v_c \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^2U_5. \tag{24}$$

Equating both ends (1,3) and (2,3) of formula (24), it can get

$$\begin{cases} -d_2 e_4 - d_4 e_3 = Fd_2 - He_2, \\ d_3 d_4 - e_3 e_4 = -Hd_2 - Fe_2. \end{cases} \tag{25}$$

It combines formula (21) and formula (25) to get

$$\beta_4 = \beta_{234} - \beta_2 - \beta_3. \tag{26}$$

Equating both sides (3,1) and (3,2) of formula (24), it gets

$$\frac{e\beta_5}{d\beta_5} = \frac{d\beta_1 q_b - e\beta_1 q_a}{d\beta_1 o_b - e\beta_1 o_a}. \tag{27}$$

Letting $d\beta_1 q_b - e\beta_1 q_a = I$ and $d\beta_1 o_b - e\beta_1 o_a = Z$, get

$$\beta_5 = r \tan 2(I, Z). \tag{28}$$

To sum up, the inverse solution of the five-degree-of-freedom robot is obtained. In practical work, the most

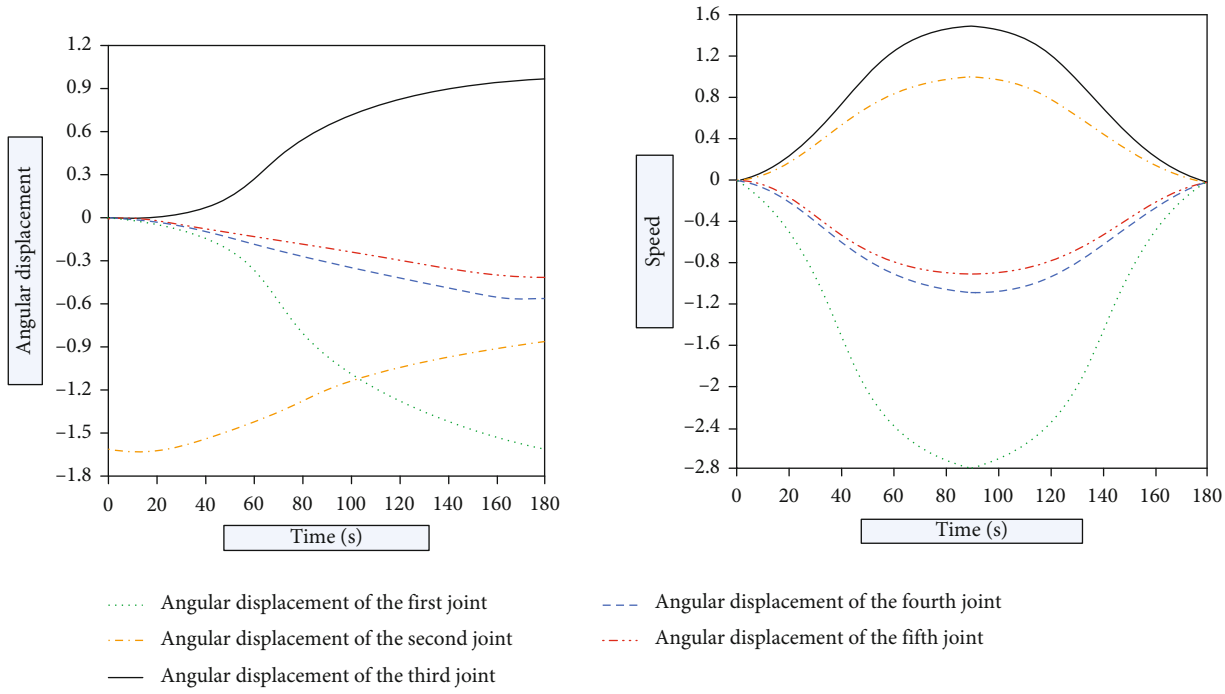

Figure 6: Changes in the angle and speed of each joint of the picking robot.

suitable solution can be selected based on the method that conforms to the shortest path, the shortest time-consuming, and the least energy-consuming method to improve the efficiency of the robot [23].

4. Robot Displacement Trajectory Tracking Experiment

4.1. Neural Network Control. The displacement trajectory tracking of the picking robot controlled by the neural network is shown in the following figure:

It can be concluded from Figure 5 that the actual trajectory of the picking robot can completely coincide with the expected trajectory in 0.7 seconds.

In the actual operation process, there may be errors in the joint operation of the robot, which will cause small fluctuations [24]. But this will not affect the tracking performance.

The angle and velocity changes of each joint are shown in Figure 6.

It can be seen from Figure 6 that the joints of the picking robot move independently. In the process of moving, whether the speed suddenly changes will have a greater impact on the working effect and performance of the robotic arm. If there is a sudden change in speed, the robotic arm will be impacted and vibrated accordingly. This will affect the stability of the manipulator and reduce the efficiency [25–27]. The change curve of each joint angle and velocity reflected in Figure 6 is continuous and smooth. The speed of the start node and the stop node is both 0, and there is no sudden change. This shows that the robot under the control of neural network runs smoothly and has good performance. It is feasible to use neural network control to plan and track the displacement of the robot.

4.2. Displacement Trajectory Tracking Control Experiment. The experiment was divided into two phases. In the first stage, the line segment connecting D_1 and D_2 in the planning space is the desired trajectory of the end of the third axis of the picking robot. Assuming that the motion time is 0.83 seconds, the parameters of the controller are $C_D = C_H = 7.0$ and $\varphi = [14, 0.4, 7]^t$, and the iterative process selects a round trip between D_1 and D_2.

The arc connecting G_1 and G_2 in the second-stage planning plane is the expected trajectory of the end of the second axis of the picking robot. Assuming that the movement time is 5.28 seconds, the parameters of the controller are also $C_G = C_H = 7.0$ and $\varphi = [14, 0.4, 7]^t$, and the back and forth between G_1 and G_2 is used as an iterative process. Then, the trajectory error information is obtained, and the error is substituted into the control law for calculation, and the iteration coefficient and compensation torque of the first iteration process are obtained. It is then transmitted to the controller, which operates the robot for a second movement [28]. Repeating the above operations continuously can obtain the error data of each movement, as shown in Figure 7.

The variation of each axis error of the picking robot trajectory movement is shown in Figure 7. It can be seen from the figure that the error fluctuates with the increase of time. In the first axis, when the time is 300 s and 500 s, the error is the largest; in the second axis, when the time is 100 s, 300 s, 500 s, and 700 s, respectively, the error is maximum. Because of the leading term $\lambda_{-1} = 0$ of the first iteration, the first iteration curve can be viewed as a trajectory error curve without

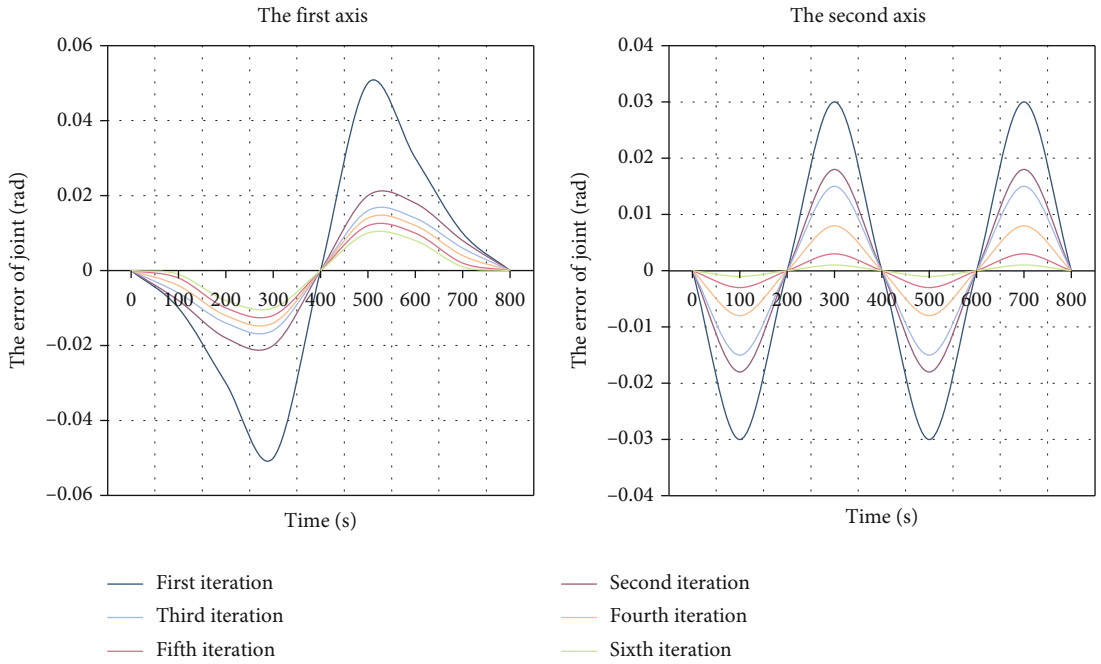

Figure 7: The variation of each axis error of picking robot linear trajectory motion.

Table 2: Linear trajectory error.

Initial maximum error		Maximum error after six iterations	Percentage of initial error
First axis	3.8×10^{-2}rad	1.8×10^{-3}rad	2.2%
Second axis	2.3×10^{-2}rad	1.8×10^{-3}rad	3.6%
Difference	1.5×10^{-2}rad	0	1.4%

Table 3: Circular path error.

Initial maximum error		Maximum error after six iterations	Percentage of initial error
First axis	2.1×10^{-2}rad	1.3×10^{-2}rad	77.4%
Second axis	6.1×10^{-3}rad	5.3×10^{-3}rad	78.8%
Difference	14.9×10^{-3}rad	7.7×10^{-3}rad	3.4%

algorithmic guidance. However, the axis error of the subsequent iteration curves decreases with the increase of the number of iterations. The axis error is greatly reduced after 6 iterations, as shown in Tables 2 and 3.

In the test, the reducer of the first axis used a harmonic reducer, which made the friction torque smaller [29]. Therefore, the curve of the first axis is gentler than that of the second axis. The second axis is a linear motion that is transmitted to the ball screw through a multistage gear train reducer and a synchronous belt. However, the gear train reducer has the disadvantage of uneven friction, so it shows a trend of fluctuation.

Figure 8 shows the error curve of each axis of the picking robot's arc trajectory motion. It can be seen from the figure that the data shows a fluctuating trend. Both the first axis and the second axis reach the maximum error when the time is 2000 s, and the error of the second axis reaches a small peak at 1000 s. Since the arc motion is limited by the centripetal acceleration, the convergence speed of the arc orbit is slower than that of the linear orbit. So the running speed will drop, which makes the iterative effect less obvious than the linear motion [30]. The specific data table is shown in Table 3. The circular arc trajectory movement has a sudden change at 2000 ms, which corresponds to the corner G_2 of the trajectory.

The experimental results show that the adaptive iterative learning control algorithm can well complete the trajectory tracking task. With the increase of the number of iterations, the trajectory error can be reduced to a certain range and has better tracking effect for the case of large initial error.

4.3. Displacement Trajectory Optimization of Picking Robot. In order to improve the operating efficiency of the robot and ensure that the robot can complete the task in a short time, the trajectory of the robot can be optimized to reduce the impact force the robot receives during operation [31].

It starts from the initial population and selects the best individual in the current cluster based on the evaluation function values of all corresponding individuals. It uses intersection and mutation to generate new subclusters [32], which evolve the clusters into better regions to find the optimal solution. The main calculation process is as follows:

(1) First, assign all parameters, including the number of individuals, the number of variables, the frequency

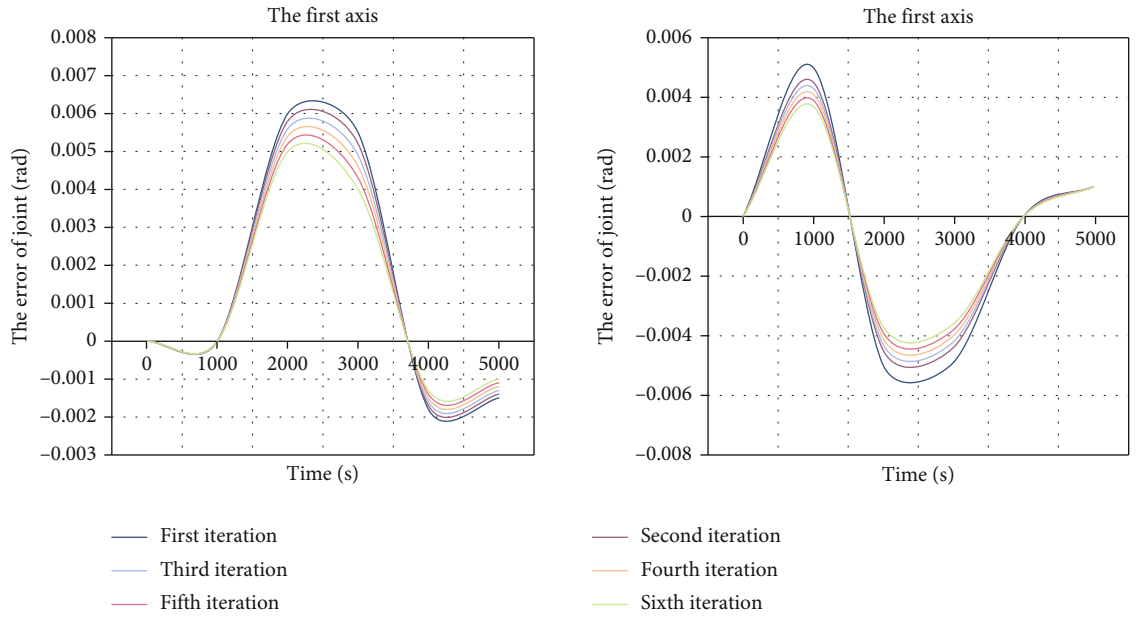

Figure 8: The variation of each axis error of the picking robot's arc trajectory motion.

of crossover, the probability of mutation that may occur, and the algebra used to end the operation

(2) It creates an area scanner and determines the interval to which it belongs. The interval to which a variable belongs is determined by the constraints of finding the variable for a particular problem

(3) In the interval to which the variable belongs, convert all individual codes that exist in the solution space, so that the solution space is transformed into a genetic space

(4) It determines the initial cluster, randomly generates a parent cluster containing N individuals, and sets the N individuals as the starting point, and uses the neural network and D-H modeling method to start the operation. Then, let S be the last terminated genetic algebra

(5) The initial cluster is brought into the calculation, each fitness function value is obtained, and the individual with relatively excellent fitness value is selected after comparison

(6) The cluster is subjected to genetic operations, such as selection, mutation, crossover, and recombination, so that the cluster p(e) is transformed into the next generation cluster p(e+1)

(7) After a series of operations, the individual fitness values in the optimal solutions of all clusters are obtained, and the optimal solutions of each cluster are judged. The operation is terminated if the termination condition is met

The specific process is shown in Figure 9.

Letting the time interval be $T_x, x = 1, 2, \cdots, n$, the obtained time series comparison table before and after optimization is shown in Table 4.

The optimized picking robot can reach the set point of the movement trajectory in a short time. The total exercise time was shortened by 6.84 seconds from 14.19 seconds to 7.35 seconds. There is no deviation between the must-pass points after optimization and those before optimization. The picking robot can complete the motion trajectory accurately and without error. The optimized robot can also better cope with the impact on the movement, reduce the impact of the impact on the movement and picking operations, and improve the operation and work efficiency [33].

5. Discussion

This paper is dedicated to the application of artificial intelligence technology to study the displacement trajectory tracking control of agricultural picking robots. It not only roughly describes the neural network, D-H modeling method and the forward and reverse motion of the manipulator, and combining artificial intelligence with the research of displacement trajectory tracking control of picking robot, radial basis function neural network is used to guide the establishment of algorithm model, but also is a new attempt to track and control the displacement trajectory of the picking robot. By modeling the robotic arm and analyzing the movement trajectory of the robot, the displacement trajectory of the robot can be tracked and controlled, and the operation and work efficiency of the picking robot can be improved.

Through the analysis of this paper, it shows that artificial intelligence has profound significance in the trajectory tracking control of robots. The use of agricultural picking robots

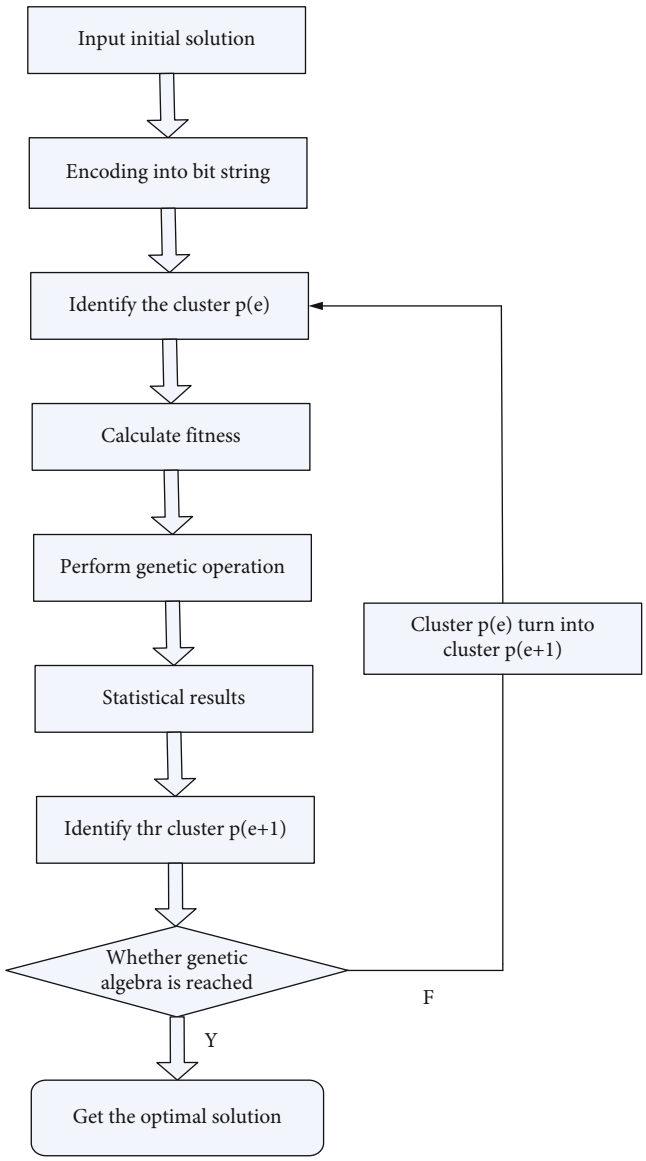

Figure 9: Flowchart of GA algorithm optimization.

Table 4: Time series comparison table before and after optimization.

Interval time (s)	Before optimization	After optimization
T_1	5.13	1.76
T_2	4.2	1.43
T_3	0.89	0.66
T_4	0.71	0.80
T_5	0.52	0.37
T_6	2.74	2.33

under the guidance of artificial intelligence technology to carry out operations such as picking, handling, and packaging of agricultural products can save labor costs, realize agricultural modernization, and improve the total output value of agriculture and the entire social economy, thereby promoting the rapid development of society and economy. In this paper, the motion simulation model of the robot is established through the neural network method, the D-H modeling method, and the forward and reverse motion analysis of the manipulator. It plans, tracks, and optimizes the motion trajectory of the robot, and the running error of the robot is greatly reduced through experimental analysis. And the total time of the optimized robot movement is shortened by 6.84 seconds, which can better cope with the impact during operation, improve work efficiency, and complete work tasks.

6. Conclusions

Through the analysis of this paper, the following conclusions are drawn: (1) The actual trajectory of the picking robot under the control of the neural network can completely coincide with the expected trajectory in 0.7 seconds. The running process of the robot is relatively stable, and the performance is better. The neural network has a good effect on the tracking control of the displacement trajectory. (2) It applies the D-H modeling method to the modeling of the manipulator and solves and analyzes the forward and inverse motion equations of the manipulator. Then, the displacement trajectory of the robot is planned, and experiments are carried out to simulate the displacement trajectory of the robot, and the tracking and control are carried out. The experimental results verify that the forward and reverse motion of the manipulator under the guidance of artificial intelligence technology is reasonable. The position and velocity of the end of the robotic arm changes smoothly. And the axis error is greatly reduced after 6 iterations. (3) The displacement trajectory optimization under the guidance of artificial intelligence can improve the operation efficiency of the robot and ensure the completion rate of the robot in a short time. And the optimized total motion time is shortened by 6.84 seconds, which ensures high efficiency and ensures that the displacement trajectory will not be deviated. (4) This paper makes a certain contribution to the research on the algorithm of the tracking control of the displacement trajectory of the agricultural picking robot. But there are also shortcomings. The robot will generate friction during the movement process, and this friction force is more complicated. It is very important to conduct an in-depth analysis of the friction force of the robot in the future research. In the actual operation process, the robot will encounter some obstacles such as branches and trunks during the operation. The situation and displacement changes that will occur when encountering obstacles also need further study.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declared no potential conflicts of interest with respect to the research, author- ship, and/or publication of this article.

Acknowledgments

This work was supported by the Special Fund Project for Rural Revitalization Strategy of Bureau of Science and Technology of Huizhou Municipality (No. 2021SC040202005).

References

[1] M. Ou, H. Sun, Z. Zhang, and S. Gu, "Fixed-time trajectory tracking control for nonholonomic mobile robot based on visual servoing," *Nonlinear Dynamics*, vol. 108, no. 1, pp. 251–263, 2022.

[2] X. Ma, Y. Zhao, and Y. Di, "Trajectory tracking control of robot manipulators based on U-model," *Mathematical Problems in Engineering*, vol. 2020, Article ID 8314202, 10 pages, 2020.

[3] H. Peng, F. Li, J. Liu, and Z. Ju, "A Symplectic instantaneous optimal control for robot trajectory tracking with differential-algebraic equation models," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 5, pp. 3819–3829, 2020.

[4] S. A. Khalilpour, R. Khorrambakht, H. Damirchi, H. D. Taghirad, and P. Cardou, "Tip-trajectory tracking control of a deployable cable-driven robot via output redefinition," *Multibody System Dynamics*, vol. 16, no. 4, pp. 1–28, 2020.

[5] F. Wang, Z. Q. Chao, L. B. Huang, H. Y. Li, and C. Q. Zhang, "Trajectory tracking control of robot manipulator based on RBF neural network and fuzzy sliding mode," *Cluster Computing*, vol. 16, no. 7, pp. 1–11, 2017.

[6] Y. Bensafia, S. Ladaci, K. Khettab, and A. Chemori, "Fractional order model reference adaptive control for SCARA robot trajectory tracking," *International Journal of Industrial and Systems Engineering*, vol. 30, no. 2, pp. 138–156, 2018.

[7] N. He, L. Qi, R. Li, and Y. Liu, "Design of a model predictive trajectory tracking controller for mobile robot based on the event-triggering mechanism," *Mathematical Problems in Engineering*, vol. 2021, Article ID 5573467, 13 pages, 2021.

[8] P. Li, G. Bao, X. Fang, and L. Zhang, "Adaptive robust sliding mode trajectory tracking control for 6 degree-of-freedom industrial assembly robot with disturbances," *Assembly Automation*, vol. 38, no. 3, pp. 259–267, 2018.

[9] M. Keyvan, R.-J. Mohammad, and H. K. Abdollah, "A stable analytical solution method for car-like robot trajectory tracking and optimization," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 42–50, 2020.

[10] N. A. Khan, O. I. Khalaf, C. A. T. Romero, M. Sulaiman, and M. A. Bakar, "Application of intelligent paradigm through neural networks for numerical solution of multiorder fractional differential equations," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 2710576, 16 pages, 2022.

[11] G. Wang and J. Zhou, "Dynamic robot path planning system using neural network," *Journal of Intelligent Fuzzy Systems*, vol. 40, no. 2, pp. 3055–3063, 2021.

[12] H. Eschmann, H. Ebel, and P. Eberhard, "Trajectory tracking of an omnidirectional mobile robot using Gaussian process regression," *Automatisierungstechnik*, vol. 69, no. 8, pp. 656–666, 2021.

[13] C. Santos, F. Espinosa, E. Santiso, and D. Gualda, "Lyapunov self-triggered controller for nonlinear trajectory tracking of unicycle-type robot," *International Journal of Control, Automation and Systems*, vol. 18, no. 7, pp. 1829–1838, 2020.

[14] R. Randis, "Uji eksperimetal trajectory tracking pada robot penjinak bom," *Jurnal Integrasi*, vol. 11, no. 1, pp. 33–36, 2019.

[15] J. Peng, W. Xu, T. Yang, Z. Hu, and B. Liang, "Dynamic modeling and trajectory tracking control method of segmented linkage cable-driven hyper-redundant robot," *Nonlinear Dynamics*, vol. 101, no. 1, pp. 233–253, 2020.

[16] G. Li, F. Liu, A. Sharma et al., "Research on the natural language recognition method based on cluster analysis using neural network," *Mathematical Problems in Engineering*, vol. 2021, Article ID 9982305, 13 pages, 2021.

[17] J. Zhao, J. Huang, R. Wang, H. R. Peng, and S. Ji, "Investigation of the optimal parameters for the surface finish of k9 optical glass using a soft abrasive rotary flow polishing process," *Journal of Manufacturing Processes*, vol. 49, pp. 26–34, 2020.

[18] J. Zhao, J. Huang, Y. Xiang et al., "Effect of a protective coating on the surface integrity of a microchannel produced by micro-ultrasonic machining," *Journal of Manufacturing Processes*, vol. 61, pp. 280–295, 2021.

[19] H. Chen, Y. Shang, and K. Sun, "Multiple fault condition recognition of gearbox with sequential hypothesis test," *Mechanical Systems and Signal Processing*, vol. 40, no. 2, pp. 469–482, 2013.

[20] H. Chen, L. Fang, D. L. Fan, W. Huang, and L. Zeng, "Particle swarm optimization algorithm with mutation operator for particle filter noise reduction in mechanical fault diagnosis," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 3, 2019.

[21] M. Batliner, F. Breitenecker, and A. Krner, "ARGESIM benchmark C11 'SCARA robot' with extended trajectory tracking control," *SNE Simulation Notes Europe*, vol. 31, no. 1, pp. 43–51, 2021.

[22] L. Ye, G. Xiong, C. Zeng, and H. Zhang, "Trajectory tracking control of 7-DOF redundant robot based on estimation of intention in physical human-robot interaction," *Science Progress*, vol. 103, no. 3, p. 003685042095364, 2020.

[23] A. O. Pizarro-Lerma, V. Santibanez, R. Garcia-Hernandez, and J. V. Chnin, "Sectorial fuzzy controller plus feedforward applied to the trajectory tracking of robot manipulators," *IFAC-Papers OnLine*, vol. 53, no. 2, pp. 9918–9923, 2020.

[24] H. Hu, S. Xiao, and H. Shen, "Modified linear active disturbance rejection control for uncertain robot manipulator trajectory tracking," *Mathematical Problems in Engineering*, vol. 2021, Article ID 8892032, 13 pages, 2021.

[25] A. S. Andreev and O. A. Peregudova, "On output feedback trajectory tracking control of an omni-mobile robot," *IFAC-Papers OnLine*, vol. 52, no. 8, pp. 37–42, 2019.

[26] H. Zhu, H. Wei, B. Li, X. Yuan, and N. Kehtarnavaz, "Real-time moving object detection in high-resolution video sensing," *Sensors*, vol. 20, no. 12, p. 3591, 2020.

[27] G. H. Choi, H. Ko, W. Pedrycz, A. K. Singh, and S. B. Pan, "Recognition system using fusion normalization based on morphological features of post-exercise ECG for intelligent biometrics," *Sensors*, vol. 20, no. 24, p. 7130, 2020.

[28] R. Hassan, F. Bendary, K. Elserafi, A. Ghanem, and M. Soliman, "Comparative study methods of trajectory tracking control for robot manipulator (Dept. E)," *Bulletin of the Faculty of Engineering Mansoura University*, vol. 40, no. 4, pp. 1–12, 2020.

[29] J. Rubio, P. C. Francisco, E. García, C. F. Juarez, and J. Lopez-Gomez, "Trajectory tracking of the robot end effector for the minimally invasive surgeries," *International Journal of*

Business Intelligence and Data Mining, vol. 16, no. 1, pp. 66–88, 2020.

[30] B. Ren, Y. Wang, and J. Chen, "Trajectory-tracking-based adaptive neural network sliding mode controller for robot manipulators," *Journal of Computing and Information Science in Engineering*, vol. 20, no. 3, pp. 1–23, 2020.

[31] S. F. Hasan and H. M. Alwan, "Design of hybrid controller for the trajectory tracking of wheeled mobile robot with Mecanum wheels," *Journal of Mechanical Engineering Research and Developments*, vol. 43, no. 5, pp. 400–414, 2020.

[32] M. J. Mahmoodabadi and N. Nejadkourki, "Trajectory tracking of a flexible robot manipulator by a new optimized fuzzy adaptive sliding mode-based feedback linearization controller," *Journal of Robotics*, vol. 2020, Article ID 8813217, 12 pages, 2020.

[33] A. Welabo and G. Tesfamariamr, "Trajectory tracking control of UR5 robot manipulator using fuzzy gain scheduling terminal sliding mode controller," *Journal of Mechatronics and Robotics*, vol. 4, no. 1, pp. 113–135, 2020.