Driving Path Tracking Strategy of Humanoid Robot Based on Improved Pure Tracking Algorithm

Yuan Wei, Haoran Li, Zhigang Zhou, Dongdong Chen

Abstract—To improve the path-tracking accuracy of humanoid robot driving, this paper proposes a path-tracking strategy for humanoid robot vehicle going based on an improved pure tracking algorithm. The strategy takes into account the influence of vehicle speed and path curvature on the foresight distance. The vehicle's speed is adjusted according to the path curvature changes, enabling dynamic adjustments of the foresight distance and steering wheel turning angle. A controller is designed using a fuzzy control algorithm to minimise the path-tracking error. The vehicle's lateral and heading errors are taken as the fuzzy controller inputs, while the steering wheel angle compensation is taken as the output. The sum of the feedforward and compensation of the steering wheel angle is used as the control amount of the vehicle's steering wheel angle. To reduce the computational amount and save calculation time, the movement primitives of robot driving motion are extracted and connected to design the robot driving motion module. This module takes the vehicle parameters as input for the motion model and outputs the corresponding robot-related joint angles. The humanoid robot NAO and a miniature electric vehicle are the test platforms. In the Linux system, the robot's NAOqi system and the vehicle's main controller communicate through the same local area network, enabling the robot NAO to drive the vehicle in real-time. Experimental results demonstrate that the proposed strategy significantly improves the path-tracking accuracy of humanoid robot vehicle driving compared to traditional algorithms.

Index Terms—humanoid robot, path tracking, pure tracking algorithm, foresight distance, fuzzy control

I. Introduction

WITH the continuous development of science and technology, the control precision of robots has been dramatically improved, and their application is more and more extensive [1], [2]. The success of the Urban Challenge DARPA [3] and the advancement of artificial intelligence [4] have raised expectations for robot-driving cars operating in urban environments. Robot-driving cars are a new idea of installing robots into cabs instead of human drivers to achieve

Yuan Wei is an Associate Professor of the School of Vehicle Engineering, Henan University of Science and Technology. Luoyang 471000, China (e-mail:9905675@haust.edu.cn)

Haoran Li is a Postgraduate student of the School of Vehicle Engineering, Henan University of Science and Technology. Luoyang 471000, China (Corresponding author, e-mail: lihaoranbg@163.com).

Zhigang Zhou is an Associate Professor of the School of Vehicle Engineering, Henan University of Science and Technology. Luoyang 471000, China (e-mail:hnmczzg@163.com)

Dongdong Chen is a Postgraduate student of the School of Vehicle Engineering, Henan University of Science and Technology. Luoyang 471000, China (e-mail: chendongdong30996@163.com).

uncrewed vehicles in dangerous and harsh situations [5-8]. Moreover, integrating robotics and vehicles provides an expanded research platform, enabling humanoid robots to be utilized in a broader range of applications.

At present, there is a lot of research on robot-driving technology and path-tracking technology. Antonio Paolillo et al. [9], [10] proposed a sensor-based reactive framework for implementing the central part of the overall driving task. Cherubini et al. [11] summarized an autonomous driving project involving a musculoskeletal humanoid robot, leveraging the hardware and software of the robot Musashi to enable pedal and steering wheel operation. Carlos Carranco et al. [12] proposed a robot software architecture for the autonomous driving of electric vehicles, successfully achieving autonomous navigation. Chen et al. [13], [14] proposed a hierarchical coordination control method based on fuzzy logic theory to achieve coordinated control and accurate speed tracking for driving test cycles under various operating conditions. Chen et al. [15] proposed using a PI controller as the primary controller and a low-pass filter to mitigate abrupt changes in the steering angle. Zhang et al. [16] proposed a path-tracking algorithm for agricultural machinery based on an improved pure-tracking model. Although the linear tracking accuracy improved, it was unsuitable for agricultural machinery to perform coursework. Yang et al. [17] designed an intelligent vehicle tracking control system based on angle compensation to improve path-tracking accuracy.

This paper develops a kinematic model between the robot's dual arm kinematics and the vehicle steering system. It solves the functional relationship between the path trajectory, the steering wheel turning angle of the vehicle, and the joint angle of the robot's dual arms. A path tracking control strategy for humanoid robot vehicle driving is proposed based on an improved pure tracking algorithm. The strategy takes into account the impact of path curvature on the vehicle's driving speed. It dynamically adjusts the foresight distance based on the functional relationship between the vehicle speed and foresight distance. The foresight distance plays a crucial role in determining the steering wheel angle that the robot needs to turn while driving. Finally, the effectiveness of the strategy is experimentally verified.

The paper is organized as follows. The kinematics model of humanoid robot vehicle driving is established in Section II. Section III an improved pure tracking algorithm is proposed, and a fuzzy controller is designed. Section IV describes the vehicle driving process of a humanoid robot. In section V, the performance of the proposed strategy is evaluated and analyzed. Section VI, Conclusions and future research are given.

II. MATHEMATICAL MODELLING

A. Robot NAO

The humanoid robot NAO is used as the robot experiment platform. According to the humanoid robot NAO's dual arm structure, the link frame of both arms is established using D-H parameters. The configuration is shown in Fig. 1.
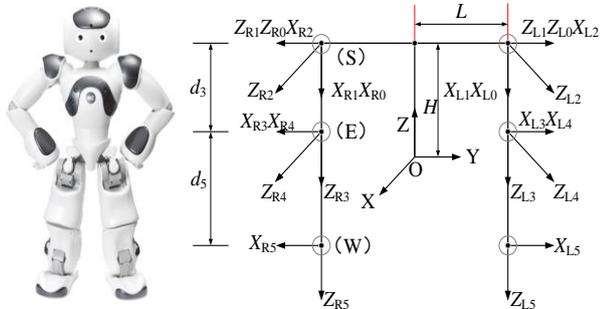
Fig. 1. D-H linkage coordinate system

In the cartesian coordinate system, Point O is the origin. The point S is the shoulder joint, point E represents the elbow joint, and Point W represents the wrist joint. d_3 is the length of the upper arm, d_5 is the length of the lower arm, l_h is the length of the hand, L is the offset of the shoulder joint relative to the origin on the Y-axis, and H is the offset of the shoulder joint relative to the origin on the Z-axis.

As the NAO hand is fixedly connected to the wrist, the arm end coordinates are established at the center of the hand. The robot's dual arm D-H parameters are shown in Tables 1 and 2.

TABLE I
D-H PARAMETERS OF THE LEFT ARM OF THE NAO

i	$\alpha_{i-1}/(°)$	a_{i-1}/mm	d_i/mm	θ_{li}	range/(°)
1	-90	0	0	θ_{l1}	-119.5~119.5
2	90	0	0	θ_{l2}	-18~76
3	90	0	d_3	θ_{l3}	-119.5~119.5
4	-90	0	0	θ_{l4}	-88.5~-2
5	90	0	d_5	θ_{l5}	-104.5~104.5
T	0	0	l_h	0	

TABLE II
D-H PARAMETERS OF THE RIGHT ARM OF THE NAO

i	$\alpha_{i-1}/(°)$	a_{i-1}/mm	d_i/mm	θ_{ri}	range/(°)
1	90	0	0	θ_{r1}	-119.5~119.5
2	-90	0	0	θ_{r2}	-76~18
3	-90	0	d_3	θ_{r3}	-119.5~119.5
4	90	0	0	θ_{r4}	2~88.5
5	-90	0	d_5	θ_{r5}	-104.5~104.5
T	0	0	l_h	0	

The forward kinematics of the NAO arms are solved according to the D-H parameters of the arms. The homogeneous coordinate transformation matrix T_i represents the pose of link i relative to the previous link, and the arm joints of NAO are all revolute joints without prismatic joints, so the expression of T_i can be given as follows:

$$T_i = \text{Rot}(Z, \theta_i)\text{Trans}(0, 0, d_i)\text{Trans}(a_i, 0, 0)\text{Rot}(X, \alpha_i)$$

$$= \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where Trans is the translation transformation matrix, and Rot is the rotation transformation matrix. $C\theta_i$ is the $\cos(\theta_i)$, $S\theta_i$ is the $\sin(\theta_i)$, $C\alpha_i$ is the $\cos(\alpha_i)$, and $S\alpha_i$ is the $\sin(\alpha_i)$.

The coordinate transformation matrix for the left and right arms from wrist to shoulder are as follows:

$$_6^1T_L = T_1 T_2 \text{Rot}(Y, -pi/2) T_3 T_4 T_5 \text{Rot}(Y, pi/2) T_6 \quad (2)$$

$$_6^1T_R = T_1 T_2 \text{Rot}(Y, pi/2) T_3 T_4 T_5 \text{Rot}(Y, -pi/2) T_6 \quad (3)$$

Fig.1 shows the shoulder joint that is rotated relative to the robot's center about the X axis by 90 degrees and translated H units in the Z axis and L units in the Y axis. The coordinate transformation matrix from the center of the robot to the left and right shoulders are shown as follows:

$$T_{LSO} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L \\ 0 & 0 & 1 & H \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$T_{RSO} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -L \\ 0 & 0 & 1 & H \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$R_{LX} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$$R_{RX} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

The transformation matrices from the left and right arm end-effectors to the center of the robot are denoted by T_{Ln} and T_{Rn}, respectively. The specific expressions are as follows:

$$T_{Ln} = T_{LSO} R_{Lx6}^{\ 1}T_L = \begin{bmatrix} L_{11} & L_{12} & L_{13} & P_{LX} \\ L_{21} & L_{22} & L_{23} & P_{LY} \\ L_{31} & L_{32} & L_{33} & P_{LZ} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

$$T_{Rn} = T_{RSO} R_{Rx6}^{\ 1}T_R = \begin{bmatrix} R_{11} & R_{12} & R_{13} & P_{LX} \\ R_{21} & R_{22} & R_{23} & P_{LY} \\ R_{31} & R_{32} & R_{33} & P_{LZ} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

where L_{mn} is the functional relationship in the rotation matrix. R_{mn} is the functional relationship in the rotation matrix.

The hands of the NAO are simple toggle grippers that do not affect the kinematics calculations, so the calculation of

the hand is ignored. In this paper, the method described in the literature [18] is used to solve the robot's joint angles.

The joint angles of the left arm of NAO are solved as follows:

$$\theta_{L1} = \arctan\left(\frac{(d_5+l_h)L_{32}+P_{LZ}-H}{P_{LX}+(d_5+l_h)L_{12}}\right)$$

$$\theta_{L2} = \arctan\left(\frac{L-(d_5+l_h)L_{22}-P_{LY}}{(d_5+l_h)(L_{32}S_{L1}+L_{12}C_{L1})+S_{L1}(P_{LZ}-H)+C_{L1}P_{LX}}\right)$$

$$\theta_{L3} = \arctan\left(\frac{L_{12}S_{L1}-L_{32}C_{L1}}{-L_{12}C_{L1}S_{L2}-L_{22}C_{L2}-L_{32}S_{L1}S_{L2}}\right) \quad (10)$$

$$\theta_{L4} = -\arccos(L_{22}S_{L2}-L_{12}C_{L1}C_{L2}-L_{32}S_{L1}C_{L2})$$

$$\theta_{L5} = \arctan\left(\frac{L_{11}C_{L1}C_{L2}-L_{21}S_{L2}+L_{31}S_{L1}C_{L2}}{L_{13}C_{L1}C_{L2}-L_{23}S_{L2}+L_{33}S_{L1}C_{L2}}\right)$$

The joint angles of the right arm of NAO are solved as follows:

$$\theta_{R1} = \arctan\left(\frac{(d_5+l_h)R_{32}-P_{RZ}+H}{P_{RX}-(d_5+l_h)R_{12}}\right)$$

$$\theta_{R2} = \arctan\left(\frac{(d_5+l_h)R_{22}-L-P_{RY}}{(d_5+l_h)(R_{32}S_{R1}-R_{12}C_{R1})-S_{R1}(P_{RZ}-H)+C_{R1}P_{RX}}\right)$$

$$\theta_{R3} = \arctan\left(\frac{R_{12}S_{R1}-R_{32}C_{R1}}{R_{12}C_{R1}S_{R2}+R_{22}C_{R2}-R_{32}S_{R1}S_{R2}}\right) \quad (11)$$

$$\theta_{R4} = \arccos(R_{12}C_{R1}C_{R2}-R_{22}S_{R2}-R_{32}S_{R1}C_{R2})$$

$$\theta_{R5} = \arctan\left(\frac{-R_{11}C_{R1}C_{R2}+R_{21}S_{R2}+R_{31}S_{R1}C_{R2}}{R_{13}C_{R1}C_{R2}-R_{23}S_{R2}-R_{33}S_{R1}C_{R2}}\right)$$

B. Vehicle kinematic model

In this paper, a simplified kinematic model of a mini-electric vehicle is employed to describe the vehicle's motion. The robot sits in the position, and its hands are fixed in the gripping position during operation. In the process of path tracking, the vehicle's speed is low, and the fluctuation of the driving road is slight. Then, the miniature electric car can be simplified as a 2-Dofs vehicle model, as shown in Fig.2.
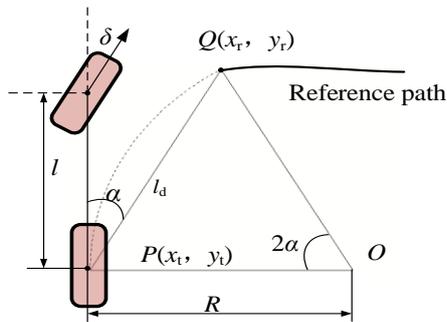
Fig. 2. Two-degree-of-freedom model of the vehicle

α is the orientation angle of the foresight point in the vehicle coordinate system, R is the turning radius, l_d is the foresight distance, $P(x_t, y_t)$ is the rear wheel center coordinates, $Q(x_r, y_r)$ is the target point, l is the vehicle wheelbase, δ is the front wheel turning angle.

According to the geometric relation in Fig. 2, the front wheel angle is:

$$\delta = \arctan\left(\frac{2l\sin(\alpha)}{l_d}\right) \quad (12)$$

Neglecting the dynamics of the steering mechanism, the wheel turning angle is proportional to the steering wheel turning angle φ; the relationship is as follows:

$$\varphi = \frac{\delta}{\mu} \quad (13)$$

where μ is the coefficient between the front wheel angle and the steering wheel angle.

C. Driving motion model

Research has shown that a 9-point 3-point grip is the most reasonable, correct, and safe grip [19]. The relative position relationship between the steering wheel center and the robot is shown in Fig. 3.
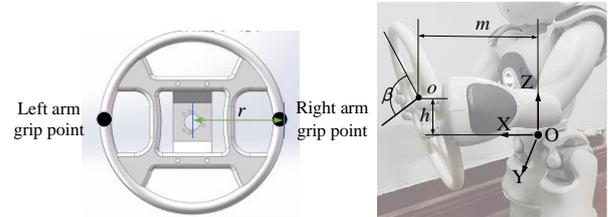
（a）Steering wheel grip point （b）The relative position of NAO and steering wheel
Fig. 3. The relative position relationship between the steering wheel centre and the robot

c and h represent the offsets of the steering wheel center relative to the robot center in the X-axis and Z-axis directions, respectively. β represents the rotation angle of the steering wheel axis relative to the Y-axis.

Take T_0 to denote the positional matrix of the center of the steering wheel concerning the center of the robot; the expression is as follows:

$$T_0 = \text{Trans}(c,0,h)\text{Rot}(Y,\beta)$$

$$= \begin{bmatrix} \cos\beta & 0 & \sin\beta & c \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & h \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

During the rotation of the steering wheel, the transformation matrix of the left-hand grip point concerning the center of the robot is T_L, and the transformation matrix of the right-hand grip point concerning the center of the robot is T_R. The expression is as follows:

$$T_L = T_0\text{Rot}(Z,\varphi)\text{Trans}(Y,r)\text{Rot}(X,\frac{\pi}{2})\text{Rot}(Z,\frac{\pi}{2}) \quad (15)$$

$$T_R = T_0\text{Rot}(Z,\varphi)\text{Trans}(Y,-r)\text{Rot}(X,-\frac{\pi}{2})\text{Rot}(Z,-\frac{\pi}{2}) \quad (16)$$

Since the grasping position of both hands is fixed, the relationship between the grasp point on the steering wheel and the arm end-effector can be expressed as follows:

$$T_{LN} = T_L \quad (17)$$

$$T_{RN} = T_R \quad (18)$$

Therefore, the corresponding values of matrix T_L and T_R can be substituted into equations (10) and (11) to calculate the joint angles.

When the robot turns the steering wheel, a closed loop state is formed between the dual arm and the steering wheel. Once the motion trajectory of the two grasp points is determined, the constraint relationship between the dual arm is determined. The schematic diagram of the cooperation under the dual-arm constraint is shown in Fig. 4.
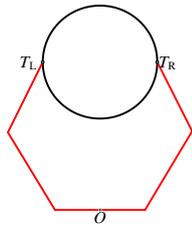
Fig. 4. The diagram of dual-arm cooperation.

The closed circular lines in Fig. 4 represent the dual-arm motion trajectories. According to the steering wheel's structural parameters, both hands' grip points are always constant during the turning of the steering wheel. The matrix constraint relationship between the dual-arm end effectors of the robot is as follows：

$$T_L = T_R \text{Trans}(0,0,2r)\text{Rot}(Y,\pi)\text{Rot}(Z,\pi) \qquad (19)$$

When the robot is driving, it should be ensured that the end positions of the arms satisfy the constraint relationship all the time. If they do not, feedback adjustment will be performed. The specific details and methods of the feedback adjustment are not elaborated in this paper.

The robot regulates the driving speed of the vehicle through adjustments in the ankle angle. The schematic diagram between the accelerator pedal and the sole surface of the foot is shown in Fig. 5.
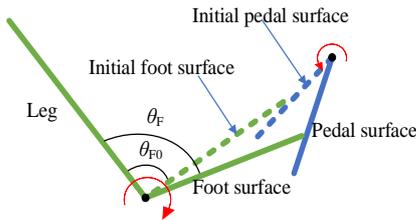
Fig. 5. Gas pedal operation diagram

Through experiments, it is found that there exists a linear correlation between the ankle joint angle and the vehicle's traveling speed. The relationship is shown as follows:

$$\theta_F = kv + \theta_{F0} \qquad (20)$$

where θ_F represents the ankle joint angle, k represents the proportional relationship, and θ_{F0} represents the ankle joint angle in the initial state.

III. PATH TRACKING CONTROLLER

A. The function of path curvature and velocity

In contrast to the traditional algorithm where the foresight distance l_d is set as a constant [20], in this paper, some factors were considered to determine the foresight distance. Firstly, the vehicle travel speed v during path tracking is determined based on the path curvature variation. Secondly, the foresight distance is determined by establishing a functional relationship between the vehicle speed and the foresight distance. Finally, the steering wheel turning angle required for the robot to drive is calculated.

Previous research [21] has demonstrated that the degree of the desired path curvature change has an impact on the ve-

hicle speed. To ensure the stability of the driving, the speed of the vehicle is reduced when the path becomes more curved. The diagram of path curvature change is shown in Fig. 6.
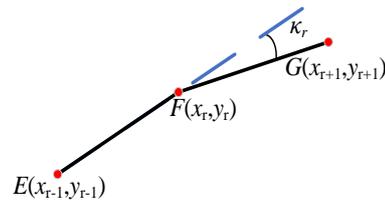
Fig. 6. Schematic diagram of the change in path curvature

$E(x_{r-1},y_{r-1})$, $F(x_r, y_r)$, and $G(x_{r+1},y_{r+1})$ represent three adjacent waypoints on the given path, respectively. Based on a given path point, the calculation process for k_r is shown as follows:

$$k_r = \pi - \arccos \frac{(x_r - x_{r-1})(x_r - x_{r+1}) + (y_r - y_{r-1})(y_r - y_{r+1})}{\sqrt{(x_r - x_{r-1})^2 + (y_r - y_{r-1})^2}\sqrt{(x_{r+1} - x_r)^2 + (y_{r+1} - y_r)^2}} \qquad (21)$$

According to k_r, the degree of the desired path curvature change can be calculated using the following formula:

$$V = \sum_{r=1}^{n} |k_r| \qquad (22)$$

where n is the number of waypoints included in each calculation.

The vehicle speed v varies with the V. When V is smaller, the vehicle speed v is higher, and vice versa. Therefore, the vehicle speed can be expressed as:

$$v = p \bullet \frac{(V_{max} - V)}{V_{max} - V_{min}} + q \qquad (23)$$

where p is the curvature adjustment parameter, q is the preset constant and $p,$ and q can be determined based on actual vehicle tests. V_{max} is the maximum degree of change in curvature; V_{min} is the minimum degree of curvature change.

B. Improved pure tracking algorithm

The foresight distance l_d plays a crucial role in determining the accuracy of path tracking. The selection of the foresight distance involves a trade-off between smooth trajectory tracking and maintaining stability during curve turning. A longer foresight distance results in a smoother trajectory. However, it can lead to under-steering or over-steering at the curve turning point. On the other hand, a shorter foresight distance prompts frequent adjustments of the front wheel angle based on the close foresight point, which can negatively impact vehicle stability. The vehicle's speed significantly influences the choice of the foresight distance. A longer foresight distance is preferred at higher vehicle speeds to avoid abrupt turns. Conversely, a shorter foresight distance is selected at lower vehicle speeds to ensure accurate tracking of curved paths.

Based on the above analysis and the literature [22], a function between the foresight distance and vehicle speed is proposed, with the following relationship:

$$l_d = Av^2 + Bv + C \qquad (24)$$

where A is the vehicle braking distance, B is the speed regulation factor, and C is half the vehicle's wheelbase.
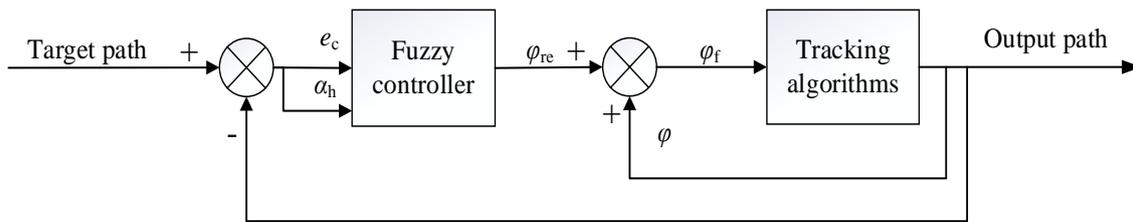
Fig. 7. Control logic diagram of fuzzy controller

C. Fuzzy controller

The pure tracking control algorithm will have tracking errors in the process of tracking the path. A fuzzy controller based on a fuzzy control algorithm is designed to improve the accuracy of path tracking further. The vehicle's lateral error e_c and heading error α_h are taken as the input of the fuzzy controller, and the compensation of the steering wheel angle φ_{re} is taken as the output. The sum of the φ_{re} and φ is used as the control amount φ_f of the vehicle's steering wheel angle. The specific control logic diagram is shown in Fig. 7.

The range of lateral error e_c is -10 ~10cm, the range of heading error α_h is -5 ~ 5°, the range of front wheel angle compensation φ_{re} is -3 ~ 3°, and the quantisation factor is 1. The fuzzy processing is shown in Table 3.

TABLE III
FUZZY PROCESSING

Designation	The basic theory of domain	Fuzzy language set
EC	[-10cm,10cm]	NB NM NS ZO PS PM PB
α_h	[-5°,5°]	NB NM NS ZO PS PM PB
φ_{re}	[-3°,3°]	NB NM NS ZO PS PM PB

The basic principle of fuzzy control rules for automatic compensation of steering wheel angle is as follows. When the lateral error is positive and the heading error is positive, the steering wheel angle compensation amount should be output to a negative value. When the lateral and heading errors are negative, the steering wheel angle compensation should be output to a positive value. The fuzzy control rules are shown in Table 4.

TABLE IV
FUZZY RULES

φ_{re}		α_h						
		NB	NM	NS	ZO	PS	PM	PB
	NB	PB	PB	PM	PS	PS	ZO	ZO
	NM	PB	PB	PM	PS	PS	ZO	NS
	NS	PM	PM	PM	PS	ZO	NS	NS
e_c	ZO	PM	PM	PS	ZO	NS	NM	NM
	PS	PS	PS	ZO	NS	NS	NM	NM
	PM	PS	ZO	NS	NM	NM	NM	NB
	PB	ZO	ZO	NB	NM	NM	NB	NB

Triangular affiliation functions are chosen as the inputs and outputs. The control surface of the adaptive fuzzy controller derived from the fuzzy rules in Table 4 is shown in Fig. 8.
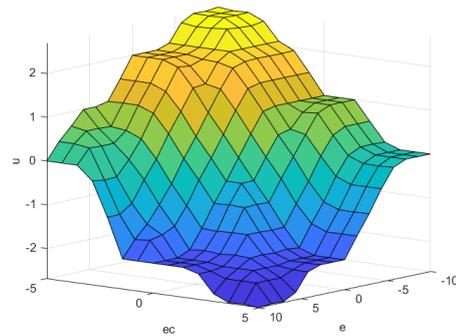
Fig. 8. Control surface of the adaptive fuzzy controller

The lateral error e_c and the heading error α_h are calculated as shown in Fig. 9.
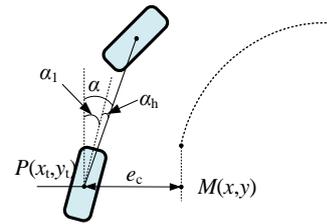
Fig. 9. Error calculation diagram

α_1 is the actual front view, and $M(x, y)$ represents the shortest vertical distance coordinate point on the ideal path from the tracking path.

The lateral error formula is as follows:

$$e_c = \sqrt{(x_t - x)^2 + (y_t - y)^2} \qquad (25)$$

The heading error formula is as follows:

$$\alpha_h = \alpha - \alpha_1 \qquad (26)$$

The relationship formula between the steering wheel angle φ_f, the actual steering wheel angle φ, and the steering wheel angle compensation φ_{re} is shown as follows:

$$\varphi_f = \varphi + \varphi_{re} \qquad (27)$$

IV. HUMANOID ROBOT DRIVING EXPRESSION

A. Data acquisition

NAOqi is the main program running on the robot, which completes the overall control of the NAO. The NAOqi architecture is a programming framework for programming the robot NAO, which allows communication, programming,
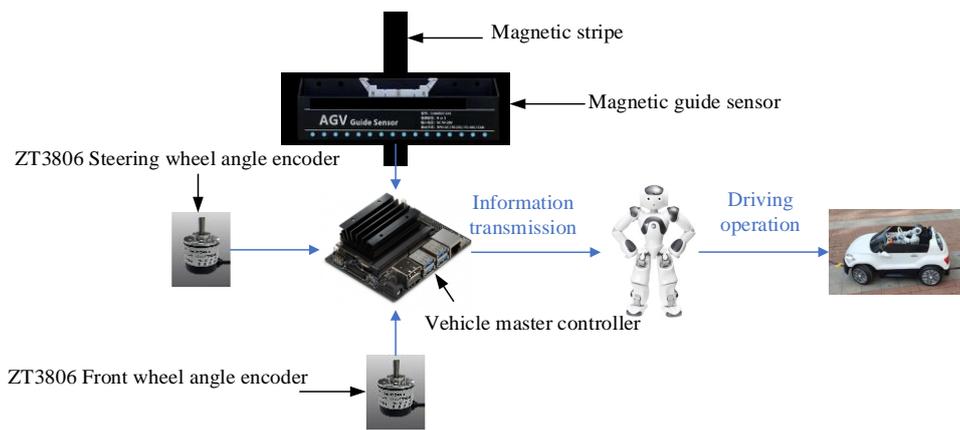
Fig. 10. Hardware connection diagram

and information sharing between different modules. Users can develop code under Windows, Mac, or Linux systems and call through C++, Python, Net, and other languages. Under the Linux system, the NAO and the vehicle central controller are communicated on the same network, and the data transmission frequency is 50HZ. The vehicle central controller stores the vehicle position, speed, and other information obtained by relevant sensors, and the kinematics of the NAO are calculated based on these data to determine the joint angle. In the NAOqi architecture, Python programming is used to complete the real-time control of NAO. The overall hardware connections are shown in Fig. 10.

B. Motion representation of movement primitives

To reduce the calculation of the robot's driving motion and save the calculation time, a modular approach is employed in the design of the robot's motion. The function of different parts of the robot body is analyzed and divided, and the related function modules are designed. By combining these functional modules, the driving movement of the robot is constituted to meet the driving requirements.

Before the robot initiates its driving motion, it undergoes initial adjustments to prepare for movement. During the driving phase, the robot primarily focuses on controlling the direction and speed of the vehicle. Therefore, the driving motion of the robot is divided into the initialization control module, direction control module, and speed control module. In the initialization control module, an initialization movement primitive is extracted to determine the initial angle of the joint of the robot's whole body. In the direction control module, two kinds of movement primitive are removed, which are used to control the change of the joint angle of both arms of the robot to control the vehicle's driving direction. In

the speed control module, a movement primitive is extracted, which is used to control the vehicle's driving speed by changing the joint angles of the leg.

Based on the above analysis, four kinds of movement primitives are extracted. According to the extracted movement primitive, the corresponding program module are written in Python, as shown in Table 5. When the robot's NAOqi system is configured in the Python environment, the NAO robot can be controlled in real-time through the input command of the Python terminal. The flow of the application of the movement primitives is shown in Fig. 11.

TABLE V
MOVEMENT PRIMITIVES

Body parts	Schematic diagram	Object of action
Whole body		Initial state
Left arm		Steering wheel left operating point.
Right arm		Steering wheel right operating point
Right foot		Accelerator pedal
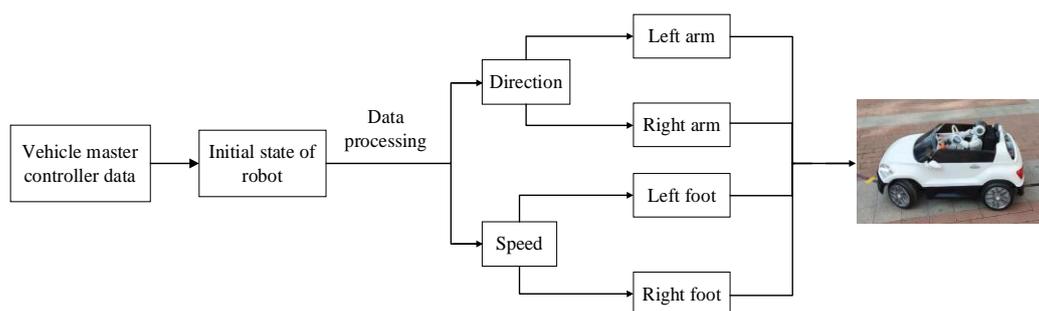
Fig. 11. Movement primitives application process

Fig. 13. Experimental platform

C. Humanoid robot driving process

According to the given waypoints, the improved algorithm proposed in this paper can be used to let the robot drive the vehicle for path tracking. The improved pure tracking algorithm is shown in Algorithm 1.

The flow chart for humanoid robot vehicle driving path tracking is shown in Fig.12.
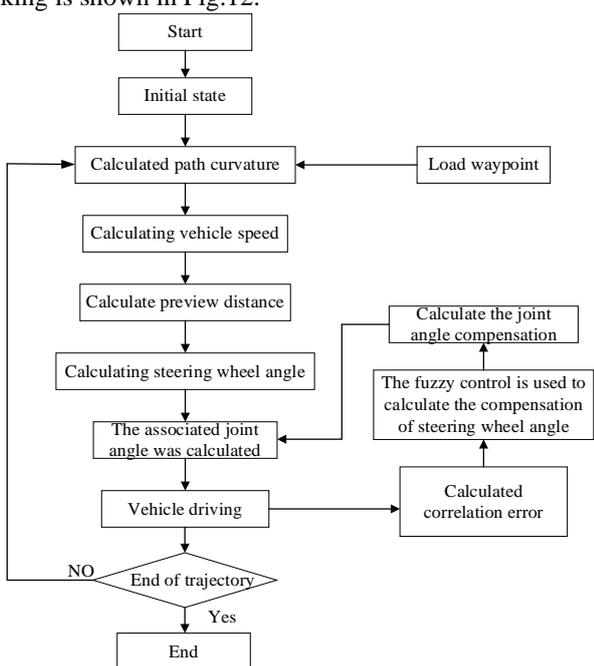
Fig. 12. The flow chart for humanoid robot vehicle driving path tracking

V. SIMULATION AND EXPERIMENTAL VALIDATION

A. Experimental platform

The experimental platform is shown in Fig. 13. The UI overview diagram of the vehicle central controller in the Linux system is shown in Fig. 14.

Fig. 14. Overview of the data of each sensor

The relevant parameters are shown in Table 6.

TABLE VI
RELEVANT PARAMETERS

Symbol	Unit	Value	Symbol	Unit	Value
L	m	0.098	μ		2
H	m	0.1	n		5
d_3	m	0.09	p		0.6
d_5	m	0.05055	q		0.4
l_h	m	0.058	k		12
c	m	0.15	A	m	0.25
h	m	0.09	B		0.4
r	m	0.1	C		0.35
l	m	0.7	R_{min}	m	2.4
β	$\circ$	30	v_{max}	m/s	1.5

B. Experimental setup

A simulation test is conducted using MATLAB to validate the effectiveness of the proposed improved pure tracking algorithm. The simulation test compares the performance of

the proposed enhanced pure tracking algorithm with the traditional pure tracking algorithms. The two given desired paths are shown in Fig. 15.

（a）Desired path 1
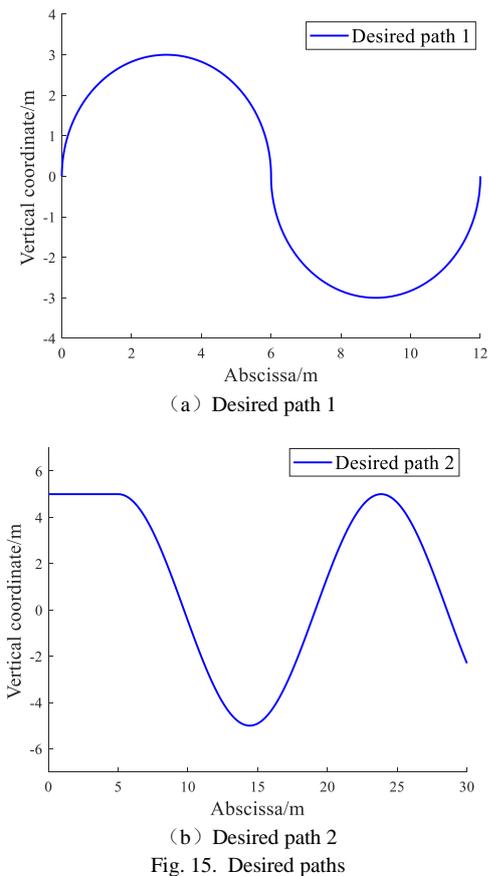
（b）Desired path 2
Fig. 15. Desired paths

In the simulation experiments conducted using the traditional pure tracking algorithm, the vehicle's initial position on the first path is set to (0, 0), and the vehicle's initial position on the second path is set to (0, 5). For desired trajectory 1, it was observed that a foresight distance of 0.7m and a speed of 0.8m/s resulted in higher accuracy. Meanwhile, for desired trajectory 2, a foresight distance of 0.8m and a speed of 0.65m/s yielded better accuracy. So, the l_d of the desired trajectory 1 was set to 0.7m, the l_d of the desired trajectory 2 was set to 0.8m, and the speed was set to 0.8m/s and 0.65m/s, respectively, in the comparison experiment.

C. Simulation experiments

The given two paths are utilized to conduct simulation experiments comparing the proposed strategy with the traditional pure tracking algorithm. The simulation results show the effectiveness of the proposed strategy. The simulation results and local magnification are shown in Fig.16. The tracking results of the proposed strategy for the two desired paths are better than the traditional pure tracking algorithm. The lateral errors of the two algorithms vary with vehicle speed, are shown in Fig. 17. In straight sections of the desired path 2, the vehicle is controlled to maintain a higher and consistent speed. A section of the desired path 1 with constant curvature with fixed curvature, the vehicle is controlled to maintain a lower and consistent speed. In variable curvature sections of the desired path 2, the control vehicle adjusts its speed in accordance with changes in path curvature. As the curvature gradually increases, the speed decreases. As the curvature gradually decreases, the control vehicle's speed

increases. When different curvatures are combined within sections, the speed is adjusted based on multiple path points at their junction. The variation of steering wheel angle compensation using the proposed method is demonstrated in Fig.18. Steering wheel angle compensation is within the specified range. The statistical results are shown in Table 7.

（a1）Desired path 1

（a2）Local magnification

（b1）Desired path 2
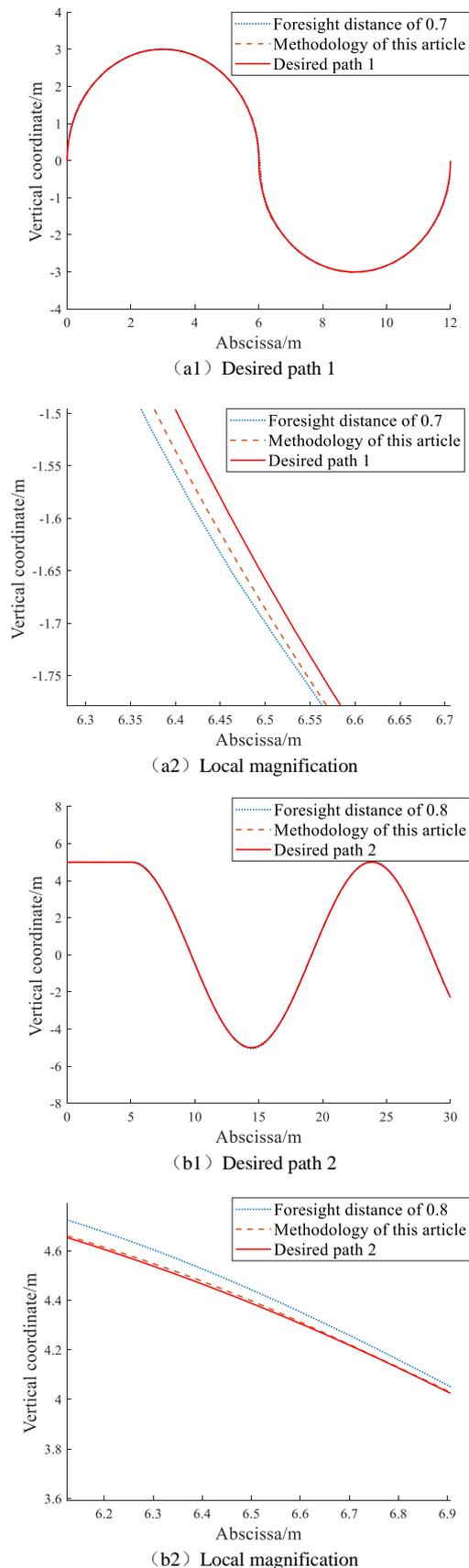
（b2）Local magnification
Fig. 16. Comparison of path tracing results

（a）Desired path 1

（b）Desired path 2
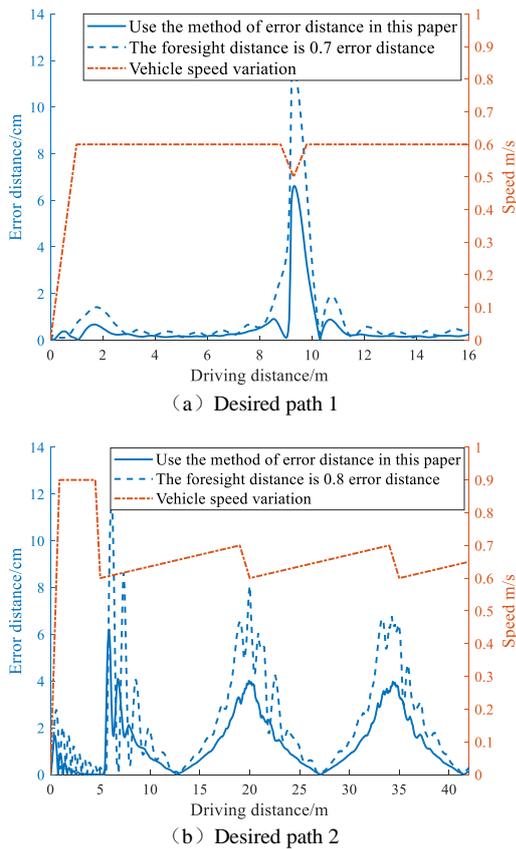
Fig. 17. The absolute values of lateral errors vary with vehicle speed

（a）Desired path 1

（b）Desired path 2
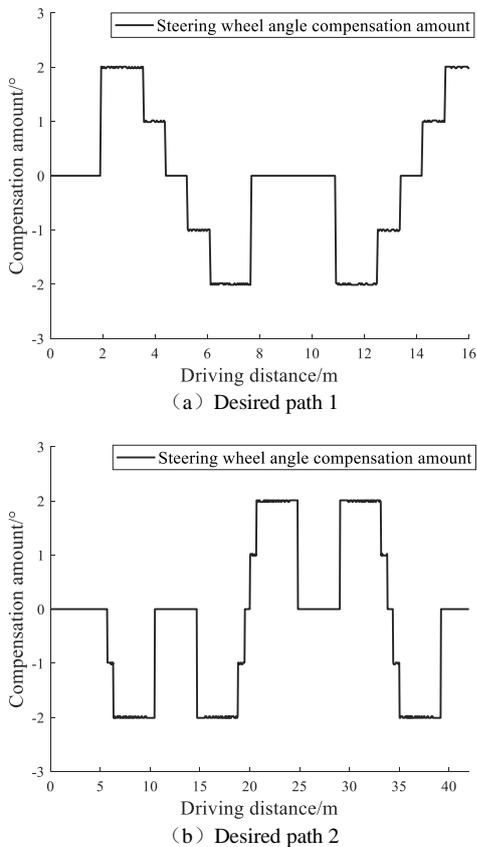
Fig. 18. Steering wheel angle compensation curve

As shown in Fig. 16 and 17, several key findings can be deduced. In the initial stage, due to the increase in vehicle speed, the foresight distance increases from the minimum value and is smaller than the fixed value of the traditional

algorithm, and the lateral error is smaller than the traditional algorithm. When the speed of the linear segment is constant, the two errors are close, the lateral error is smaller than the traditional algorithm, and the vehicle speed and foresight distance of the proposed method are more significant than the traditional algorithm. When entering the fixed curvature line segment, the vehicle speed decreases, the foresight distance becomes smaller, and the lateral error of the vehicle is smaller than that of the traditional algorithm. When entering a section with variable curvature, the curvature of the path changes more, and the foresight distance decreases. The lateral error of the proposed method is smaller than that of the traditional algorithm.

TABLE VII
STATISTICAL RESULTS OF THE LATERAL ERROR IN THE PATH

Desired path	Foresight distance	Standard Deviation /cm	Mean of the absolute value of The error/cm
1	This papper	1.0155	0.6017
	0.7	1.2040	1.1558
2	This papper	1.9150	1.4366
	0.8	2.1663	2.1502

By analyzing the simulation data in the first path, the method proposed in this paper reduces the mean value of lateral error by 47.94% compared with the traditional pure tracking algorithm. In the second path, the method proposed in this paper reduces the mean value of lateral error by 33.19% compared with the traditional pure tracking algorithm. The comparison results once again prove the advantages of the proposed strategy.

D. Robot driving experiment

The humanoid robot NAO and the mini electric vehicle were used to verify the proposed method, and the data output from the vehicle controller was classified and sorted. The path tracking experiment process of NAO driving a vehicle on desired path 1 is shown in Fig.20. The figure shows four processes in which NAO drives the vehicle to turn right, left and left. The path tracking data was analyzed and processed, and the lateral error changes of the two paths were obtained, as shown in Fig 21. Due to the influence of experimental equipment and experimental site, the error of experimental results is larger than that of simulation results. When NAO was driving the vehicle to carry out the path tracking experiment, the change in steering wheel angle was shown in Fig.22. In the section with constant curvature; the steering wheel angle fluctuates in a certain range according to the steering wheel angle compensation output by the fuzzy controller. In the variable curvature section, the steering wheel angle fluctuates greatly. The joint angles of the robot corresponding to the two paths are shown in Fig. 23 and Fig. 24. The forward kinematics is derived from the joint angle data of both arms, and the end effectors matrix of the dual-arm is obtained. According to the end effectors matrix, the dual-arm constraint relationship is verified, and the joint changes of dual-arm satisfy the dual-arm constraint relationship. Table 8 shows the statistical results of lateral errors using the proposed algorithm for the robot driving experiment. It can be seen from the table that the lateral error results meet the driving requirements of a humanoid robot.
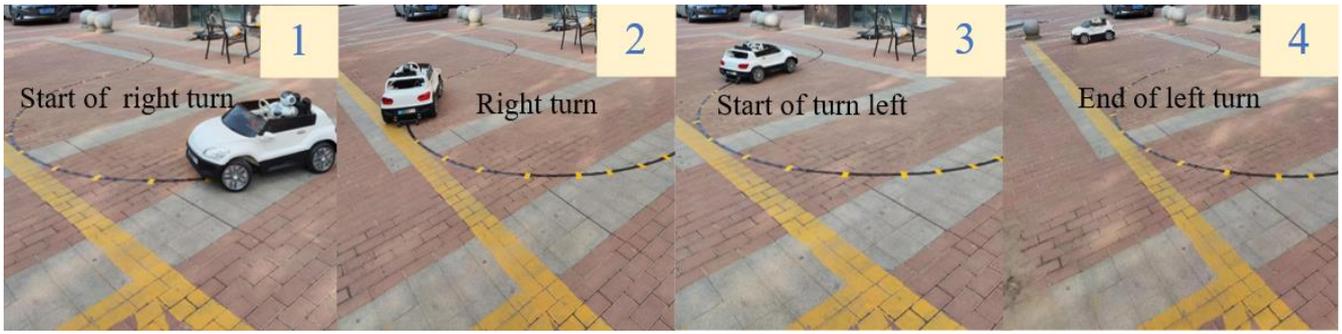
Fig. 20. Turning process
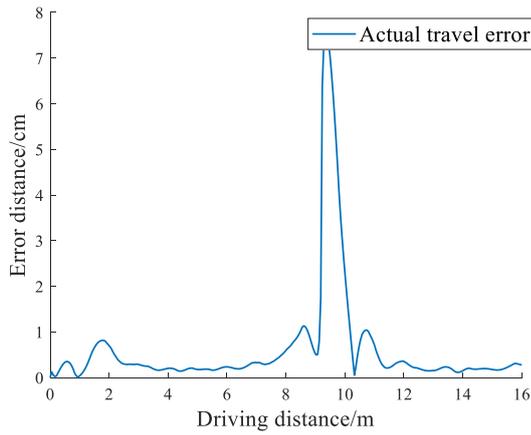
（a）Desired path 1
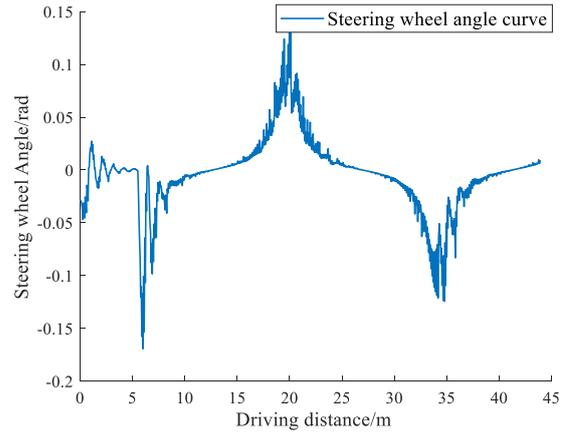
（b）Desired path 2

Fig. 22. Steering wheel turning angle
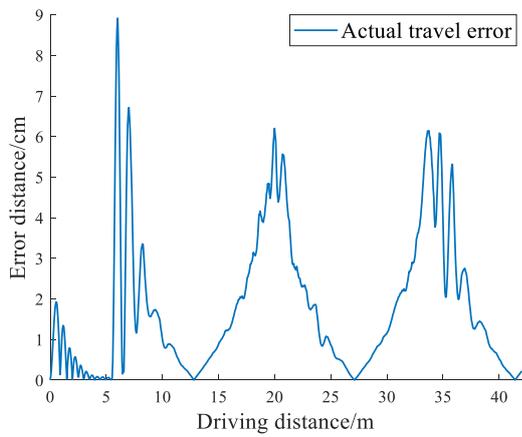
（b）Desired path 2

Fig. 21. The lateral error variation of the two paths
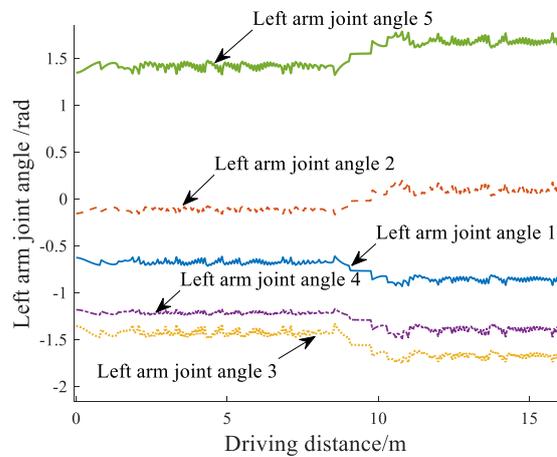
（a）Joint angle of the left arm
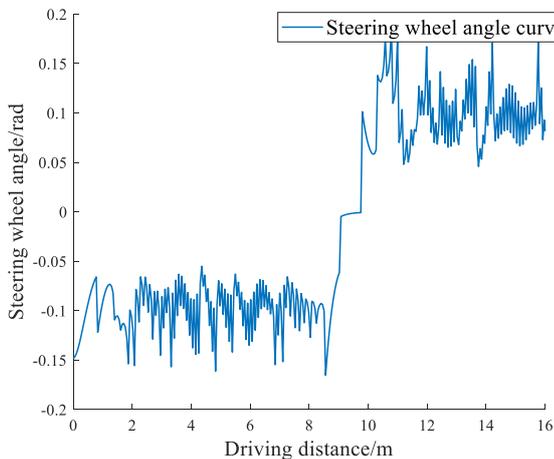
（a）Desired path 1
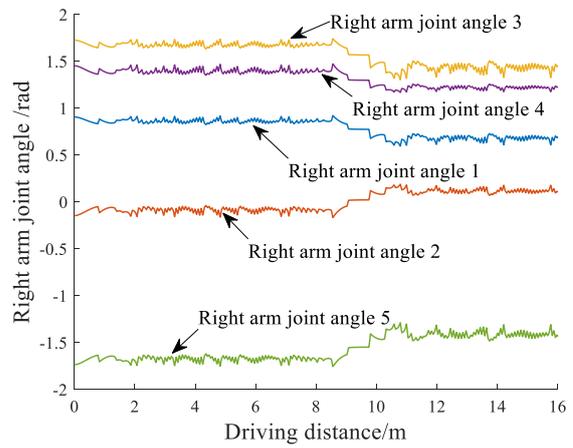
（b）Joint angle of the right arm

Fig. 23. The joint angles of both arms corresponding to the first path

（a）Joint angle of the left arm

（b）Joint angle of the right arm
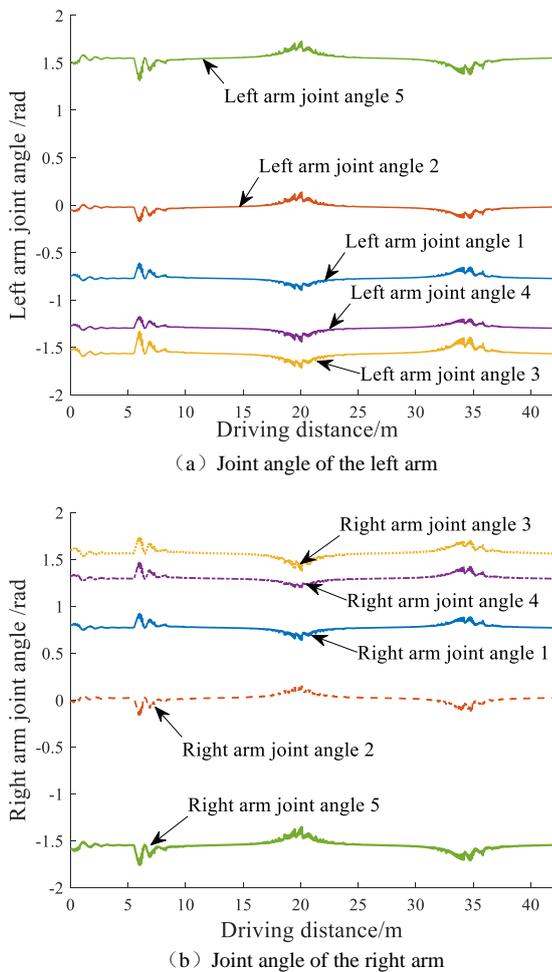
Fig. 24. The joint angles of both arms corresponding to the second path

TABLE VIII
STATISTICAL RESULTS OF THE LATERAL ERROR IN THE PATH

Desired path	Standard Deviation /cm	Mean of the absolute value of The error/cm
1	1.1910	0.7057
2	2.0623	1.9781

In the experiments, the driving error is caused by the magnetic strip arrangement error, the car's steering clearance, and the influence of the experimental environment. Comparing Fig.14 and Fig.16, the robot driving path error is slightly larger than the simulation path error, but the error is within a certain range. The lateral errors using the proposed method are smaller than those of the traditional pure tracking algorithm. In the straight path section, although the errors of the two methods are unstable, the fuzzy controller can significantly reduce the path tracking errors by compensating the steering wheel angle. In the variable curvature section, the foresight distance is adjusted in real time by analyzing the curvature change degree of the path, and the path tracking error is greatly reduced. On the road section with fixed curvature, through the coordination of fuzzy controller and improved pure tracking algorithm, the precision of path tracking is also significantly improved, making the driving route more fit to the desired path. Experiments show that the accuracy of robot driving path tracking can be enhanced by using this method, which proves the effectiveness of the proposed method and verifies the feasibility of humanoid robot driving.

VI. CONCLUSION

In order to improve the path-tracking accuracy of humanoid robot vehicle driving, a path-tracking strategy of humanoid robot vehicle driving based on the improved pure tracking algorithm is proposed. In path tracking, a fuzzy controller is designed to adjust the steering wheel turning angle in real-time through the steering wheel turning angle compensation amount to reduce the steady-state error. At the same time, in order to reduce the calculation of the robot driving motion and save the calculation time, a movement primitive motion module is established. Take the humanoid robot NAO and a miniature electric vehicle as the test platform. The experimental results show that this strategy can improve the path-tracking accuracy of humanoid robot vehicle driving compared to traditional algorithms.

In the context of the growing development of artificial intelligence, humanoid robots will be one of the most effective platforms to help humans with all their daily tasks. The study's results also provide some experience for actual robotic driving operations. Future work includes improving the accuracy and efficiency of automated two-armed driving and stability studies regarding vehicle dynamics.

REFERENCES

[1] R. Christof, H. Daniel, "A Mobile Assistive Robot for Intralogistics Applications," *Engineering Letters*, vol. 27, no.4, pp. 893-900, 2019.
[2] Y. Wang, J. Zhao, and Y. Q. Chen, "Robot Joint Trajectory Tracking Control Algorithm with Human-like Smooth Reaching Motion Profile Based on Sole Velocity Feedback," *Engineering Letters*, vol. 27, no.4, pp. 763-775, 2019.
[3] K. L. M. Buehler and S. S. (Editors), "Special issue on the 2007 DARPA Urban Challenge, part I-III," *Journal of Field Robotics*, 2008.
[4] M. H. Zhang, "Vehicle Detection Method of Automatic Driving based on Deep Learning," *IAENG International Journal of Computer Science*, vol. 50, no.1, pp86-93, 2023
[5] B. Sun, W. T. Li, H.B. Liu, et al. "Mathematical Method for Lidar-based Obstacle Detection of Intelligent Vehicle," [J]. *IAENG International Journal of Computer Science*, vol. 48, no.2, pp. 181-189, 2021.
[6] G. Chen, W. G. Zhang. "Hierarchical coordinated control method for unmanned robot applied to automotive test," [J]. *IEEE Transactions on Industrial Electronics*, vol. 63, no.2, pp. 1039-1051, 2016.
[7] N. Hirata, N. Mizutani, H. Matsui, et al. "Fuel consumption in a driving test cycle by robotic driver considering system dynamics," [C]. *International Conference on Robotics and Automation*, pp. 3374-3379, 2015.
[8] B. Alt, E. Hermann, "Second-order sliding modes control for rope winch based automotive driver robot," [J]. *International Journal of Vehicle Design*, vol. 62, pp. 147-164, 2013.
[9] A. Paolillo, A. Cherubini, M. Vendittelli, "Autonomous car driving by a humanoid robot," *Journal of Field Robotics*, vol. 35, no.2, pp. 169-186, 2018.
[10] A. Paolillo, A. Cherubini, F. Keith, M. Vendittelli, "Toward autonomous car driving by a humanoid robot: A sensor-based framework," *IEEE-RAS International Conference on Humanoid Robots*, pp. 451-456, 2015.
[11] K. Tsuzuki et al., "Toward Autonomous Driving by Musculoskeletal Humanoids: A Study of Developed Hardware and Learning-Based Software," *IEEE Robotics & Automation Magazine*, vol. 27, no. 3, pp. 84-96, 2020.
[12] Carranco C., Encalada P., Gavilanes J., Delgado G., Garcia M.V, "An Approach of a Control System for Autonomous Driving Based on Artificial Vision Techniques and NAO Robot," *Advances in Intelligent Systems and Computing*, vol. 1066, pp. 196-206, 2020.
[13] G. Chen, W. Zhang, "Hierarchical coordinated control method for unmanned robot applied to automotive test," [J]. *IEEE Transactions on Industrial Electronics*, vol. 63, no. 2, pp. 1039-1051, 2016.

[14] G. Chen, W. Zhang, X. Li, B. Yu, "Adaptive speed control method for electromagnetic direct drive vehicle robot driver based on fuzzy logic," [J]. *Measurement and Control*, vol. 52, pp. 1344-1353, 2019.

[15] Y. Chen, Y. Shan, "Optimization of pure pursuit controller based on PID controller and low-pass filter, " [C] //2018 *21stInternational Conference on Intelligent Transportation Systems(ITSC)*, 2018：3294－3299.

[16] H. Q. Zhang, G. D. Wang, Y. F. Lv, "Agricultural machinery automatic navigation control system based on improved pure tracking model, " [J]. *Transactions of the Chinese Society for Agricultural Machinery*, 2020, 51(9): 18-25.

[17] Y. Y. Yang, Z. G. He, R. C. Wang, " Path tracking control system of intelligent vehicle based on steering angle compensation," [J]. *Instrument Technique and Sensor*, 2019(5): 73-77.

[18] Nikolaos. K, Emmanouil. O, Michail. G. L, "Complete Analytical Forward and Inverse Kinematics for the NAO Humanoid Robot, " [J]. *Journal of Intelligent & Robotic Systems*，2015，77（2）：251-264．

[19] W. H. Zhang, "How to correctly operate the Steering Wheel [J]. *Automotive and Safety*, " 2016(10): 102-104.

[20] C. C. Wu, S. X. Wu, L. Wen, "Variable curvature path tracking control for the automatic navigation of tractors, " [J]. *Transactions of the Chinese Society for Agricultural Machinery*, 2022, 38(21): 1-7.

[21] K. Zhao, Y. Zhu, M. Y. Feng , "Research on path tracking algorithm of autonomous vehicles based on multi-point sequence preview, "[J]. *Automobile Technology*, 2018(11): 1-5.

[22] J. Wang, "Autonomous ground vehicle control system for high-speed and safe operation, " [J]. *International Journal of Vehicle Autonomous Systems*, 2009, 7(1/2): 18-35.