Smart Structures and Systems, *Vol. 34, No. 2 (2024) 117-127*
https://doi.org/10.12989/sss.2024.34.2.117

A Runge-Kutta scheme for smart control mechanism with computer-vision robotics

ZY Chen [1], Huakun Wu [2], Yahui Meng[**1] and Timothy Chen[*3]

[1] *School of Science, Guangdong University of Petrochemical Technology, Maoming, Guangdong, China*
[2] *School of Computer Science, Guangdong Polytechnic Normal University, Guangzhou, Guangdong, China*
[3] *Engineering and Applied Science, California Institute of Technology, Pasadena, CA, USA*

(*Received August 5, 2020, Revised September 18, 2024, Accepted September 24, 2024*)

Abstract. A novel approach that the smart control of robotics can be realized by a fuzzy controller and an appropriate Runge-Kutta scheme in this paper. A recently proposed integral inequality is selected based on the free weight matrix, and the less conservative stability criterion is given in the form of linear matrix inequalities (LMIs). We demonstrate that this target information obtained through image processing is subjected to smart control with computer-vision robotic to Arduino, and the infrared beacon was utilized for the operation of practical illustrations. A fuzzy controller derived with a fuzzy Runge-Kutta type functions is injected into the system and then the system is stabilized asymptotically. In this study, a fuzzy controller and a fuzzy observer are proposed via the parallel distributed compensation technique to stabilize the system. This paper achieves the goal of real-time following of three vehicles and there are many areas where improvements were made. Finally, each information is transmitted to Arduino via I2C to follow the self-propelled vehicle. The proposed calculation is approved in reproductions and ongoing smart control tests.

Keywords: anthropomorphic robotics; computer vision; control algorithm; following system; Hough circle; intelligent control function; Runge-Kutta scheme

1. Introduction

For almost two decades, intelligence is a popular field, which catches many researchers' attention (see Braz-Cesar and Barros 2018, Chen *et al*. 2022a, b, Li *et al*. 2023, 2024, Jin *et al*. 2024, Shi *et al*. 2023, and the references therein). Nowadays, the computing speed of microprocessors has increased significantly, and they can handle complex calculations and quickly update information, making machine vision more widely used in various aspects, such as detection in hazardous environments, through the installation of various sensors Unmanned vehicles are used to collect information, or in the military, drones are used to detect difficult-to-reach locations and environments, and more closely related to our lives are robotic arms used in industrial production or transporting materials in factories. Autonomous self-propelled vehicles, these applications are just the tip of the iceberg of machine vision. Through different sensors and hardware combined with image processing, various tasks of different orientations can be completed today.

The following system with machine vision naturally has higher flexibility. In addition to adding additional sensors, it can also spread risks in a timely manner and complete the task more effectively, but its cost may increase as a result.

In the field of automatic control, there are many studies focusing on self-propelled vehicles (Hou *et al*. 2023, Song *et al*. 2024, Sheng *et al*. 2024, Fang *et al*. 2024). Nowadays, machine vision is mostly used for tracking control. However, obtaining the target position has different achievements in different environments and for different purposes. Method: If there are obstacles between the target object and the tracking system, direct observation cannot be made. In this case, if you want to know the location of the target object outdoors, you can find its coordinates through the GPS global positioning service. However, due to the structure of the building, Interference makes it impossible to use indoors. Therefore, in a built indoor environment, although GPS positioning cannot be used, the target object can be identified through multi-node triangulation based on the received signal strength. While there is no obvious barrier between the target and the following system, there are also various sensing methods, such as ultrasonic sensing, infrared sensing (Zhou *et al*. 2023a, b, Gao *et al*. 2023, 2024, He *et al*. 2023, 2024, Fu *et al*. 2023, Mi *et al*. 2023, Long *et al*. 2024), and machine vision (Liu *et al*. 2023, Ding *et al*. 2021, Gu and Ren 2023, Gu *et al*. 2024).

In terms of machine vision, if you want to track an object, you must first detect the object and then track the object. To identify specific objects in images, common methods include machine learning YOLO (Chen *et al*. 2023a, b, Sun *et al*. 2023, 2024a, b, Zhu 2023, Zhou *et al*. 2023a, b), which uses convolutional neural network training to predict the probability of possible objects and divides the

*Corresponding author, Mr.,
E-mail: t13929751005@gmail.com
**Co-corresponding author, Mr.,
E-mail: mengyahui@gdupt.edu.cn

ISSN: 1738-1584 (Print), 1738-1991 (Online)

input image into small squares. The grids are predicted separately, and the threshold value is used to determine whether it is a certain object category. Finally, each object and its position in the entire image is output, which has a higher accuracy and faster speed. Therefore, many following self-propelled vehicles Object recognition will be carried out through this method (Wang *et al*. 2024a, b). However, the calculation amount is still a big burden on a simple RaspberryPi (without external neural computing stick) and it is not easy to achieve real-time processing. It is suitable for computers with higher performance and target objects. Situations with more complex and diverse characteristics. In addition, color recognition is also a common method of detecting objects. It directly detects specific objects through color components in the image. The disadvantage is that the color of the object must be the initial condition of this method, which means it has great limitations in identifying objects, and it is susceptible to environmental interference, but if the tracked object is a fixed object and the environment is simple, this simple method can still have good performance. After detecting objects, you must ensure that the objects tracked in each frame of the image are the same as the previous frame, because the contours of the detected objects to be tracked may change due to environmental changes such as light sources and occlusions, or the appearance of similar objects causes interference, causing tracking to cause misjudgments. Common methods include area tracking, model tracking, contour tracking, and feature tracking (Zhou *et al*. 2024, Zheng *et al*. 2022, 2023, Zhang *et al*. 2024). Features can be roughly divided into local features, global features, Or relative changes in features in graphics. For example, using color detection objects can be judged by the overall features, that is, the area, centroid, etc. of the segmented foreground pixel blocks. This method can determine whether they are the same Detected object (Zhang *et al*. 2024).

This paper first uses binarization, morphological processing, Canny edge detection (Yin *et al*. 2024a, b, Yang *et al*. 2024), and then performs Hough Transform to extract the center and radius of the target ball for tracking. Since the traditional Hough Transform requires a large amount of Therefore, the Hough Gradient Method, also known as fuzzy theory, is used to replace the traditional Hough transform to achieve more real-time tracking. After completing the image tracking, the following of the self-propelled vehicle must be controlled. Many papers use fuzzy control for following control (Xu *et al*. 2022, 2023, Tian *et al*. 2023). However, the processing speed of the Raspberry Pi used in this paper must bear the burden of image processing and fuzzy logic at the same time. There will be poor real-time following ability under the calculation, so this paper uses traditional PID control logic to follow.

In addition, during the following process, in addition to knowing the direction of the target, distance information must also be obtained to avoid unnecessary collisions. On the host vehicle side, the radius of the target circle is known through Hough transformation. The real-world distance can be found through the radius; the vehicle must obtain additional distance information through distance sensors such as ultrasonic and infrared sensors to complete effective following.

2. Governing theory of boundary problems

The Newton law can be proved to be irrotational if the induced motion is small and periodic and initially irrotational. A single-valued velocity potential exists and can be defined by Cheng *et al*. (1997)

$$u_j = a_j^2 \frac{\partial \phi}{\partial x_j}, \quad a_j^2 = \left(\frac{\mathrm{i}}{f_j - \mathrm{i}s_j}\right), \qquad j = 1,2,$$

For a function of three variables $f(x,y)$ the lth order derivatives, mth order derivatives of the function with respect to x and y can be obtained by localization scheme as

$$\begin{aligned} f_x^{(l)}(x_i, y_j) &= \textstyle\sum_{k=1}^{N_L} A_{i,k}^{(l)} f(x_k, y_j), \\ f_y^{(m)}(x_i, y_j) &= \sum_{k=1}^{N_M} B_{j,k}^{(m)} f(x_i, y_k) \end{aligned} \tag{1}$$

where L, M are the number of grid points in the x, y direction respectively, N_L, N_M denote the count of supported reference points in the x, y direction, $A_{i,k}^{(l)}$, $B_{j,k}^{(m)}$ are the weighting coefficients. For the first order derivatives, the weighting coefficients $A_{i,k}^{(l)}$, $B_{j,k}^{(m)}$ can be determined as follows

$$A_{i,j}^{(1)} = \frac{L^{(1)}(x_i)}{(x_i - x_j) L^{(1)}(x_j)}, \quad B_{i,j}^{(\mathrm{l})} = \frac{M^{(1)}(y_i)}{(y_i - y_j) M^{(1)}(y_j)}, \tag{2}$$

but $j \neq i$

$$\begin{aligned} L^{(1)}(x_i) &= \prod_{j=\mathrm{l}, j \neq i}^{N_L} (x_i - x_j), \\ M^{(1)}(y_i) &= \prod_{j=\mathrm{l}, j \neq i}^{N_M} (y_i - y_j) \end{aligned} \tag{3}$$

Similarly, the weighting coefficients for the l-th order derivatives and m-th order derivatives of the function with respect to x and y can be obtained as

$$\begin{aligned} A_{i,i}^{(l)} &= l\left(A_{i,j}^{(l-1)} A_{i,j}^{(l)} - \frac{A_{i,j}^{(l-1)}}{x_i - x_j}\right), \\ B_{i,i}^{(m)} &= m\left(B_{i,j}^{(m-1)} B_{i,j}^{(l)} - \frac{B_{i,j}^{(m-1)}}{y_i - y_j}\right), \end{aligned} \tag{4}$$

$$A_{i,i}^{(l)} = -\sum_{j=1, j \neq i}^{N_L} A_{i,j}^{(l)}, \tag{5}$$

The above equations are the weighting coefficients of the second-order derivatives, and the higher-order derivatives can be calculated from the first-order derivatives themselves. It is worth noting that the polynomial approximation in the method is along a straight line.

Therefore, the method can be used to obtain a numerical solution of partial differential equations with higher-order accuracy. However, the main drawback of the convectional method is the problem of an ill-conditioned global matrix. With this feature, we use a concept of localization method to overcome the problem of an ill-conditioned matrix system. By using localization, the ill-conditioned matrix is greatly alleviated because the non-zero coefficient entries in the global matrix are decreased in each row. In addition, we can arbitrarily adjust the number of referenced points locally with the physical properties of each problem. On the other hand, the method cannot be directly applied to irregular domain problems. Key aspects to consider when developing such a solver are the imposition of boundary conditions on the immersed boundaries and the accurate discretization of the governing equation in the cells cut by these boundaries. A new interpolation procedure based on the finite element shape function is proposed, allowing the systematic development of a spatial discretization scheme preserving the second-order spatial accuracy of the underlying solver.

A transformation scheme can be explained in Eq. (6)

$$u\frac{\partial \phi}{\partial x}=\begin{cases}u_{i,j}\left(A_{i-3,j}^{(1)}\phi_{i-3,j}+A_{i-2,j}^{(1)}\phi_{i-2,j}+A_{i-1,j}^{(1)}\phi_{i-1,j}+A_{i,j}^{(1)}\phi_{i,j}+A_{i+1,j}^{(1)}\phi_{i+1,j}\right), & u_{i,j}>0\\ u_{i,j}\left(A_{i-2,j}^{(1)}\phi_{i-2,j}+A_{i-1,j}^{(1)}\phi_{i-1,j}+A_{i,j}^{(1)}\phi_{i,j}+A_{i+1,j}^{(1)}\phi_{i+1,j}+A_{i+2,j}^{(1)}\phi_{i+2,j}\right), & u_{i,j}=0\\ u_{i,j}\left(A_{i-1,j}^{(1)}\phi_{i-1,j}+A_{i,j}^{(1)}\phi_{i,j}+A_{i+1,j}^{(1)}\phi_{i+1,j}+A_{i+2,j}^{(1)}\phi_{i+2,j}+A_{i+3,j}^{(1)}\phi_{i+3,j}\right), & u_{i,j}<0\end{cases} \tag{6}$$

We use central differential quadrature (CDQ) scheme for diffusion components, $(\partial^2\phi/\partial^2 x)$ is discretized by the following

$$\frac{\partial^2\phi_{i,j}}{\partial x^2}=\left(A_{i-2,j}^{(1)}\phi_{i-2,j}+A_{i-1,j}^{(1)}\phi_{i-1,j}+A_{i,j}^{(1)}\phi_{i,j}+A_{i+1,j}^{(1)}\phi_{i+1,j}+A_{i+2,j}^{(1)}\phi_{i+2,j}\right) \tag{7}$$

where $A_i^{(1)}, A_i^{(2)}$ are first and second-order weighting coefficients in the x direction.

Assuming the depth averaged Navier-Stokes equations can be expressed by the SWEs with the following assumptions: (1) incompressible fluid; (2) negligible wind stress and Coriolis force; (3) hydrostatic pressure distribution; (4) small bottom slope in both x- and y-directions; and (5) uniform velocity distribution in the vertical direction. The 2D SWES in the form of continuity and the momentum equations as follows

$$\frac{\partial h}{\partial t}+\frac{\partial (hU)}{\partial x}+\frac{\partial (hV)}{\partial y}=0 \tag{8}$$

$$\frac{\partial (hU)}{\partial t}+\frac{\partial (hU^2+0.5gh^2)}{\partial x}\frac{\partial (hUV)}{\partial y}=gh(s_x-s_{fx}) \tag{9}$$

$$\frac{\partial (hU)}{\partial t}+\frac{\partial (hUV)}{\partial x}+\frac{\partial (hV^2+0.5gh^2)}{\partial y}=gh(s_y-s_{fy}) \tag{10}$$

for a Cartesian coordinate frame in which $x-$ represents the horizontal plane and $y-$ represents the vertical direction. The following parameters are used to define the dimensional quantities in the above governing equations and later physical explanations. g is the acceleration of gravity, U, V is the depth-averaged velocity in the $x-$, $y-$ direction, respectively. s_x, s_y is the bed slops, and s_{fx}, s_{fy} is the friction slope in the $x-$, $y-$ direction, respectively. These parameters are defined as

$$U=\frac{1}{h}\int_{z_b}^{h+z_b}u(z)dz;$$
$$V=\frac{1}{h}\int_{z_b}^{h+z_b}u(z)dz;$$
$$s_x=-\frac{\partial Z_b}{\partial x};\quad s_y=-\frac{\partial Z_b}{\partial y}$$
$$s_{fx}=\frac{n^2U\sqrt{U^2+V^2}}{h^{4/3}};\quad s_{fy}=\frac{n^2V\sqrt{U^2+V^2}}{h^{4/3}}$$

where u and v are the local time-averaged velocity components. Z_b is the bed elevation and n is the manning friction coefficient.

To use a high-resolution UDQ for the SWEs, the Eqs. can be expressed as

$$Q_t+[P(Q)]_x+[R(Q)]_y=\mathrm{S(Q)} \tag{11}$$

where Q_t is the conserved physical vector; Q_t are the flux vectors; and $S(Q)$denotes the source terms, respectively.

3. Fuzzy theory with Runge Kutta scheme

To achieve high overall computational efficiency, we develop a numerical scheme involving a three-step UDQ method. A three-step Runge-Kutta method is used to discretize (21) as follows

$$\begin{aligned}Q^{(1)}&=Q^{(0)}+\Delta tL\left(Q^{(0)}\right)\\ Q^{(2)}&=Q^{(1)}+\frac{\Delta t}{4}\left[-3L\left(Q^{(0)}\right)+L\left(Q^{(1)}\right)\right]\\ Q^{(3)}&=Q^{(2)}+\frac{\Delta t}{12}\left[-L\left(Q^{(0)}\right)-L\left(Q^{(1)}\right)+8L\left(Q^{(2)}\right)\right]\end{aligned} \tag{12}$$

where $L(Q)=S(Q)-[P(Q)]_x-[R(Q)]_x-[R(Q)]_y$.

We involve three-step Runge-Kutta method as follows.

$$Q_{i,j}^{n+1/3}=Q_{i,j}^{n}+\Delta t\left(L_{i,j}^{n}\right) \tag{13}$$

$$Q_{i,j}^{n+2/3}=Q_{i,j}^{n+1/3}+\Delta t\left(-3\left(L_{i,j}^{n}\right)+\left(L_{i,j}^{n+1/3}\right)\right) \tag{14}$$

$$Q_{i,j}^{n+1}=Q_{i,j}^{n+2/3}+\frac{\Delta t}{12}\left(-\left(L_{i,j}^{n}\right)-\left(L_{i,j}^{n+1/3}\right)+8\left(L_{i,j}^{n+2/3}\right)\right) \tag{15}$$

where

$$L_{i,j}^{n}=\left(S_{i,j}^{n}-\sum_{l=1}^{N_L}A_{i,l}^{(1)}P_{l,j}^{n}-\sum_{m=1}^{N_M}B_{j,m}^{(1)}R_{i,m}^{n}\right)$$
$$L_{i,j}^{n-1/3}=\left(S_{i,j}^{n+1/3}-\sum_{l=1}^{N_L}A_{i,l}^{(1)}P_{l,j}^{n+1/3}-\sum_{m=1}^{N_M}B_{j,m}^{(1)}R_{i,m}^{n+1/3}\right) \quad (16)$$
$$L_{i,j}^{n+2/3}=\left(S_{i,j}^{n+2/3}-\sum_{l=1}^{N_L}A_{i,l}^{(1)}P_{l,j}^{n+2/3}-\sum_{m=1}^{N_M}B_{j,m}^{(1)}R_{i,m}^{n+2/3}\right)$$

$$real\left[\frac{\left(n-\frac{3}{2}\right)\pi}{\left(\frac{\varepsilon_{2x}}{\varepsilon_{4x}}\right)^{\frac{1}{2}}(h-d)\frac{a_{2x}}{a_{4z}}}\right]\le real(K_{2n})\le real\left[\frac{\left(n-\frac{1}{2}\right)\pi}{\left(\frac{\varepsilon_{2x}}{\varepsilon_{4x}}\right)^{\frac{1}{2}}(h-d)\frac{a_{2x}}{a_{4z}}}\right] \qquad n\ge 1; \quad (17)$$

$$real\left[\frac{a_{2xz}\left(n-\frac{3}{2}\right)\pi}{d}\right]\le real(K_{2n})\le real\left[\frac{a_{2xz}\left(n-\frac{1}{2}\right)\pi}{d}\right] \qquad n\ge 1. \quad (18)$$

Therefore, the fuzzy controller and the fuzzy observer are given as follows:

Control Rule i:

IF $x_{R1}(t)$ is $M_{Ri1}(\alpha_m,\beta_m)$ and … and $x_{Rk}(t)$ is $M_{Rik}(\alpha_m,\beta_m)$
THEN $u_R(t)=-K_i\hat{x}_R(t),\ i=1,2,\dots,r.$

Observer Rule i:

IF $x_{R1}(t)$ is $M_{Ri1}(\alpha_m,\beta_m)$ and ...
and $x_{Rk}(t)$ is $M_{Rik}(\alpha_m,\beta_m)$
Then $\dot{\hat{x}}_R(t)=A_i(\alpha_m,\beta_m)\hat{x}_R(t)+B_i(\alpha_m,\beta_m)u_R(t)+L_i\left(y_R(t)-\hat{y}_R(t)\right),$ (19)

where $y_R(t)=D_i(\alpha_m,\beta_m)x_R(t)$, $\hat{y}_R(t)=D_i(\alpha_m,\beta_m)\hat{x}_R(t)$ and $i=1,2,\dots,r$.

Thus, the overall fuzzy controller and fuzzy observer can be written as

$$u_R(t)=-\frac{\sum_{i=1}^{r}w_i(x_R(t),\alpha_m,\beta_m)K_i\hat{x}_R(t)}{\sum_{i=1}^{r}w_i(x_R(t),\alpha_m,\beta_m)}, \quad (20)$$

$$\dot{\hat{x}}_R(t)=\frac{\sum_{i=1}^{r}w_i(x_R(t),\alpha_m,\beta_m)\{A_i(\alpha_m,\beta_m)\}\hat{x}_R(t)+\{B_i(\alpha_m,\beta_m)u_R(t)+L_i\left(y_R(t)-\hat{y}_R(t)\right)\}}{\sum_{i=1}^{r}w_i(x_R(t),\alpha_m,\beta_m)} \quad (21)$$

where

$$\hat{y}_R(t)=\frac{\sum_{i=1}^{r}w_i(x_R(t),\alpha_m,\beta_m)D_i(\alpha_m,\beta_m)\hat{x}_R(t)}{\sum_{i=1}^{r}w_i(x_R(t),\alpha_m,\beta_m)},$$
$$y_R(t)=\frac{\sum_{i=1}^{r}w_i(x_R(t),\alpha_m,\beta_m)D_i(\alpha_m,\beta_m)x_R(t)}{\sum_{i=1}^{r}w_i(x_R(t),\alpha_m,\beta_m)} \quad (22)$$

4. Smart control design and simulation examples

The fourth-generation Raspberry Pi Model B in Fig. 1 is mainly used for image processing and real-time tracking in his study. Its main computing unit is a quad-core 1.5 GHz and the memory capacity reaches 4 G. It can run more smoothly than the third-generation Raspberry Pi Model B.OpenCV is running, and its upgraded USB 3.0 can communicate with other hardware components easily and quickly. However, compared with the third generation Shumei Pi, it consumes more power. In a 5 V environment, if the input current is less than 2.5 A, an undervoltage warning will easily appear. This reduces the operating efficiency. If a camera module is used like this paper, the input current of 3 A is sufficient.

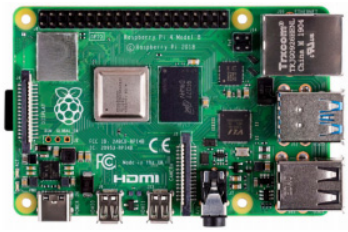

Fig. 1 Raspberry Pi 4 Model B+

Fig. 2 Photography module Raspberry Pi camera v2

Table 1 Raspberry Pi camera v2 specification sheet

Still resolution	5 Megapixels
Video modes	1080p@30fps720p@60fps 640×480p60/90fps
Sensor	Sony IMX219
Sensor resolution	3280 × 2464 pixels
Sensor image area	3.68 × 2.76 mm (4.6 mm diagonal)
Pixel size	1.12 µm × 1.12 µm
Focal length	3.04 mm
Horizontal field of view	62.2 degrees
Vertical field of view	48.8 degrees
Size	25 × twenty three × 9 mm

Fig. 3 Arduino Uno

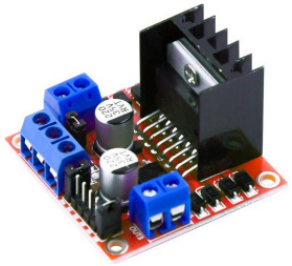

Fig. 4 Motor drive module L298N

Fig. 5 MG996R 180 degree servo motor

Image sensing module This paper uses the Raspberry Pi camera module shown in Fig. 2. It uses Sony IMX219PQ CMOS sensor, which can provide 8 million pixel still pictures, and 1080P 30 frames or 720 p 60 frames video, and uses the CSI interface (Camera Serial Interface) to transmit data, please refer to the following Table 1 for details. The Arduino Uno microprocessor in Fig. 2 is used in the main car to output PWM signals to control the motor.

L298N Fig. 3 shows a dual full-bridge motor driver with high voltage and high current output. In this paper, it is mainly used to control the speed of the DC motor so that the

Table 2 MG996R 180 degree servo motor specification

Dead Band:	0.050 ms
Control System:	Pulse Width Control
Working Frequency:	20 ms period / 50 hz (Digital Control)
Operating Voltage:	4.8 ~6 V
Operating Temperature Range:	0 to + 55 Degree C
Operating Speed (4.8v):	0.200 sec/60° degrees at no load
Operating Speed (6v)	0.160 sec/60° degrees at no load
Stall Torque (4.8v):	9.4 kg/cm
Stall Torque (6v):	11 kg/cm
Dimensions:	40.7 × 19.7 × 42.9 mm
Weight:	55 grams

Fig. 6 DC geared motor

self-propelled vehicle can move and follow. The only thing to note is that L298N uses a BJT transistor, which has a large voltage drop. Therefore, if the voltage source is only 5 V, it will be difficult to drive two DC motors, so this paper uses a voltage input of 7.4 V.

The MG996R servo motor in Fig. 4 is directly controlled through Arduino. It receives the PWM signal and can then rotate the motor to a specific angle. This paper is used to control the pitch angle of the platform where the camera module is located on the main vehicle. The purpose is to enable the main vehicle to track, which can have a certain degree of elasticity to changes in the vertical direction of the target. The following Table 2 shows its specifications.

This paper uses a reduction motor with a reduction ratio of 1:120 (Fig. 5 and Table 3). Although it is prone to errors and requires trial and error to find a reasonable PWM value, it is low-cost and suitable for use in self-made low-cost self-propelled vehicles. Each DC motor is determined by the output of L298N. The speed is controlled by the speed to achieve the purpose of going straight and turning.

In addition to the photography module introduced above, the sensing module used in this paper also includes an infrared module and an ultrasonic sensor. The infrared module includes an infrared LED and an infrared receiver. In this section, infrared, Infrared communications, transmission protocols with infrared, and ultrasonic sensors.

Infrared rays are electromagnetic waves. As long as the object is greater than absolute zero, it will radiate infrared

Table 3 Reduction ratio 1:120 reduction motor specification

Voltage	3 to 6 V
Shaft type	Double Shaft
Gear ratio	1:120
Operating speed (3.0 V)	60 ± 10% rpm (roughly) at no load
Operating speed (6.0 V)	100 ± 10% rpm (roughly) at no load

rays. Its wavelength is between visible light and microwaves, 780 nm to 1 mm (ISO 20473), and is often divided into near infrared rays (Near IR), mid infrared rays (Mid IR) and Far IR (Far IR) uses different sensors based on different wavelengths. Generally speaking, infrared light receivers only receive infrared rays with wavelengths within a specific range, so each band of infrared rays has different applications. For example, near infrared rays can be used for remote control. Mid-infrared rays can be used in gas analyzers, and far-infrared rays can be used in astronomy. The infrared LED used in this paper has a wavelength of about 940 nm and belongs to near-infrared rays. In addition, infrared rays are divided into two major categories, thermal sensors (Thermal infraded sensor) and light sensor (Quantum infraded sensor). The former has a slow response speed and has a wide response to infrared wavelengths and is often used to detect temperature and heat energy. The latter has a fast response speed and responds to a specific wavelength range and is often used for communication. The infrared sensor used in this paper is an infrared light sensor.

Infrared communication is a kind of general wireless communication. It is invisible light and can be used as short-range wireless communication. It is often used in household applications such as remote controls. However, infrared rays exist in outdoor or indoor environments, such as sunlight and fluorescent lamps, etc. all emit infrared rays. Infrared rays generally do not have hand-shake, so any infrared ray message may be received by the infrared ray receiver. Therefore, most infrared ray sensors use 38 kHz as the carrier frequency to reduce interference. The so-called carrier wave is A wave of a specific frequency (usually high frequency), the signal is sent out at the frequency of the carrier after modulation. Taking this paper as an example, the so-called carrier frequency of 38 KHz is the rapid switching of the IR LED, which is generated approximately every 26 μs. A pulse wave (Pulse) is shown in Fig. 6. Since there are few infrared light sources in the general environment that will flash in such a high-frequency mode, it can effectively avoid environmental interference. In practice, the carrier frequency can be slightly adjusted. For example, the RC5 protocol adopts 36 KHz, does not cause the infrared receiver used in this paper to be completely unable to receive information. In addition, modulation also helps to dissipate heat of the infrared LED, because compared to maintaining the same brightness and continuously emitting infrared rays, the rapid switching of modulation can make Its brief interruption, in which the proportion of switching time is called the duty cycle.

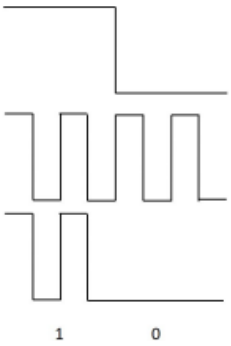

Fig. 7 Modulation diagram

Table 4 Commonly used infrared protocols and carrier frequencies

Protocol	Carrier wave freq
NEC	38 kHz
RC-5	36 kHz
RC-6	36 kHz
Panasonic	36.7 kHz
Sharp	38 kHz
Sony	40 kHz

The top is the original signal, the middle is the carrier, and the bottom is the modulated signal. General consumer products use ASK (Amplitude shift keying), that is, 0 or 1 is determined by amplitude (Table 4). When the signal is 1, the carrier wave is transmitted. When the signal is 0, the carrier wave is not transmitted and left blank (Space).

The three self-propelled vehicles in this paper are divided into one master vehicle and two slave vehicles, slave vehicle A and slave vehicle B. Both the master vehicle and slave vehicle A have infrared LEDs that emit infrared rays as beacons so that the following vehicles can follow, and as mentioned above, infrared wireless transmission uses carrier waves to resist environmental interference. In addition, in order to enable car A to follow the main car and car B to follow car A, the beacon must send out specific information to distinguish the following targets, so the transmitter There must be a common specification with the receiving end, so that the receiving end can understand the message sent by the transmitting end. This section will introduce the NEC protocol used in this paper. Use the IRremote function library on Arduino to automatically call the PWM pulse wave of 1/3 of the working cycle, that is, approximately 8.5 μ is on for the rest of the time. As shown in Fig. 7.

In the NEC agreement, PDM (Pulse Distance Modulation) is used to judge digital logic as shown in Fig. 8, that is, the time width of a unit is 560 μs, that is, a complete time unit requires about 21 pulse cycles, and under this specification, 560 μs The pulse train (PulseBurst) plus 1690 μs space, a total of 2.25 ms waveform represents Logic 1, while the 560 μs pulse train plus 560 μs space, a total of 1.12 ms represents Logic 0, because Logic 1 is determined by the position distance of the pulse wave. or 0,

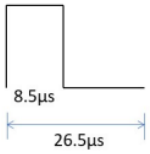

Fig. 8 38 kHz carrier frequency, pulse wave diagram of 1/3 of the working cycle

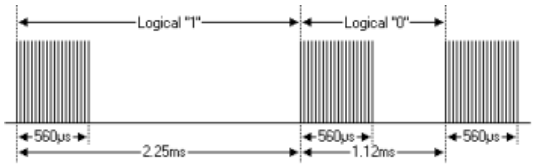

Fig. 9 Waveforms of Logic 1 and 0 under NEC specification

so it is called PDM.

A complete NEC waveform, Fig. 9, contains a 9 ms pulse train as the leader (Leader code) followed by a 4.5 ms space, telling the receiving end to start transmitting data, and the position 8 Bits (Address) and the command 8 Bits (Command) will be transmitted more. The information after one inversion totals 32 bits, allowing the receiving end to ensure that the data transmission is correct. Finally, it ends with a 560 µs pulse train. Because of the above structure, if either the sending or receiving end uses different communication protocols, it will the receiving end cannot interpret it which could be seen as Fig. 10 with information format transmitted under NEC protocol infrared transmitter.

The infrared transmitter in Fig. 11 is used to emit 38 kHz infrared rays. Due to its invisible characteristics, it is suitable as a short-distance wireless transmission tool. Since its emission angle is about 20 degrees and has directivity, it is used as a beacon in this paper. Infrared receiver for subsequent tracking and judgment.

The infrared receiver shown in Fig. 11 mainly receives infrared rays at 38 kHz. If the infrared ray frequency is far away from 38 kHz, it will not be received. In this paper, multiple infrared receivers are placed on two slave vehicles. Due to the different placement positions, the receiving

Fig. 10 Information format transmitted under NEC protocol infrared transmitter

Fig. 11 Infrared transmitter

Fig. 12 Infrared receiver

Table 5 HC-SR04 ultrasonic sensor specifications

Working voltage	5 V
Working current	15 mA
Working frequency	40 kHz
Max range	4 m
Min range	2 cm
Measuring angle	15 degree
Dimension	45 × 20 × 15 mm

range is different. The direction in which the beacon is located can be determined by which one receives the infrared message, thereby determining the slave vehicle. direction of travel. This paper uses a Nano development board to connect with an infrared sensor to process infrared reception information. If multiple infrared receivers are directly connected to the UNO board, there will be a delay, that is, multiple infrared receivers cannot receive information at the same time, which will result in the inability to make instant judgments. The direction of the vehicle ahead. This paper uses the HC-SR04 ultrasonic sensor for distance measurement to prevent the slave vehicle from getting too close and causing a collision when following. The ultrasonic sensor emits high-frequency (40 kHz) sound waves that are inaudible to the human ear. When this sound wave hits an object, it reflects the echo. The ultrasonic sensor can calculate the object based on the time difference between transmitting and receiving the echo. The distance between the sensor and the sensor is (2.1). This sensor can measure distances from 2 cm to 4 m, as shown in Table 5.

$$\text{Distance} = (\text{time difference between transmitting sound wave and receiving echo} \times \text{speed of sound})/2$$

In this paper, the Raspberry Pi uses a mobile power supply on the right side of Fig. 12, with a QC3.0 fast charging cable to achieve a 5V 3A power supply, while the Arduino microcontroller and L298N use an 18650 lithium

Fig. 13 Power supply schematic, 18650 lithium battery on the left and mobile power supply on the right

Table 6 Panasonic NCR18650B specifications

Capacity	3.4 Ah (3400 mAh)
Charging voltage	4.2 V
Energy	12.2 Wh
Internal resistance	40 mOhms (roughly)
Nominal voltage	3.6 V-3.7 V
Weight	46 g
Diameter	18.06 mm (+/- 0.03 mm)
Height	65 mm (+/- 0.03 mm)

battery on the left side of Fig. 13. Due to the low internal memory of the lithium battery. The resistor can output a higher and stable current and is more suitable as a power supply for self-propelled vehicles than alkaline or carbon-zinc batteries available in general retail stores (Table 6).

The so-called edge refers to the dividing point between two blocks in the image. This method can effectively extract the edges of objects in the image in a variety of situations. It is the most widely used edge detection algorithm, but the input image must be grayscaled in advance. The steps are as follows:

1. First perform Gaussian blur to reduce noise in the image to avoid adverse effects on the subsequent algorithm.
2. Perform gradient calculation. The purpose of this step is to determine where the edge belongs by setting the gradient threshold, that is, the brightness change at the edge should reach a certain extent.

In Canny edge detection, the Sobel Operator is used to perform a convolution operation on the image. Since the input image must be a grayscale image, the value at any pixel position is called a brightness value or grayscale value (Grayscale), since digital image information is discrete and not continuous, the Sober operation uses the difference values Gx and Gy to approximate the differential value, that is, the gradient. As long as the calculated grayscale gradient size G exceeds the set threshold, the Think of it as an edge.

The Sober operator is a 3×3 matrix that contains two directions in the mask. By convolving it with the image, the horizontal and vertical brightness gradient values can be calculated by difference instead of differential approximation. If A is the original image, Gx is the horizontal detection, Gy for longitudinal detection, the formulas are as follows: (3.9) and (3.10)

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \tag{23}$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A \tag{24}$$

gradient direction of the pixel can be obtained through Eqs. (25) and (26). At this time, as long as the gray gradient size G is greater than the threshold value, it is regarded as an edge point. This method simplifies the direction into four

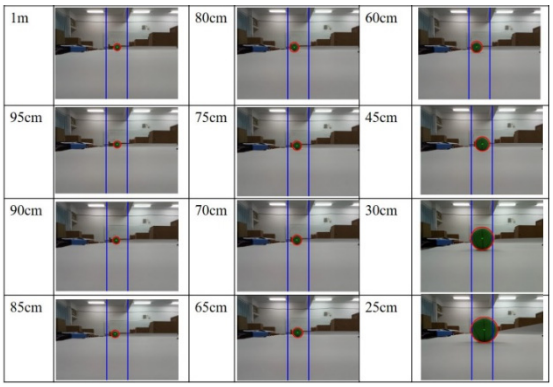

Fig. 14 Actual distance and image position, recorded every 5 cm

the directions are longitudinal, transverse, and two diagonal directions, and the gradient direction must be perpendicular to the edge.

$$G = \sqrt{G_x^2 + G_y^2} \tag{25}$$

$$\text{Angle } (\theta) = \tan^{-1}\left(\frac{G_y}{G_x}\right) \tag{26}$$

Sometimes in order to speed up the calculation of Canny edge detection in OpenCV, Eq. (25) is changed to equation (27), but generally speaking, Eq. (25) is used by default

$$G = |G_x| + |G_y| \tag{27}$$

Since the main vehicle is not equipped with any distance to the target. Therefore, experiments must be conducted to

Fig. 15 Route map of two vehicles

Fig. 16 Continuous diagram of two cars following cars

find the relationship between pixel size and distance. Figs. 14-16 shows the actual distance and the image obtained by the photography module. It shows the corresponding relationship between the actual distance and sensor, it only uses image processing to determine the pixel size of the radius of the circle. This paper uses a radius of 70 or more pixels as the decision-making interval for stopping. Because it is not possible to stop at a radius of 70 pixels very accurately, because the performance of the Raspberry Pi and the actual speed of the car body need to be taken into consideration, so it is regarded as a decision. The range is relatively appropriate.

The following is a continuous exploded diagram in Fig. 14, which is to first travel in a straight line, then turn, and then turn left after the turn is completed. Finally, it moves in a straight line and stops. The sequence of the continuous exploded diagrams is from left to right and from top to bottom. Its traveling route corresponds to Route Map 15, and the moving sequence is the same as marked on the diagram.

5. Conclusions

A novel approach that the smart control can be realized by a fuzzy controller as an auxiliary asymptotically by regulating parameters and an appropriate Runge-Kutta scheme. A fuzzy controller derived with a fuzzy Runge-Kutta type functions is injected into the system and then the system is stabilized asymptotically. In this study, a fuzzy controller and a fuzzy observer are proposed via the parallel distributed compensation technique to stabilize the system. Although this paper achieves the goal of real-time following of three vehicles, there are many areas where improvements can be made. For example, the main vehicle's target tracking method is less flexible than neural network methods such as YOLO, and can only follow a single specific target with known characteristics., so if you switch to YOLO and other methods, you can identify more objects and have more tracking options in the image, but if you use this method, you will need a stronger computing unit.

In terms of the slave vehicle's hardware, using infrared arrays, if you want to improve the performance when following turns, you may have to increase the number of infrared receivers, so that the speed control is not just about turning left and right and going straight, but can have more travel options, so that the whole following is smoother.

Acknowledgments

The authors are grateful for the research grants given to Ruei-Yuan Wang from the Projects of Talents Recruitment of GDUPT, Peoples R China under Grant NO. 2019rc098, and the research grants given to ZY Chen from the Projects of Talents Recruitment of GDUPT (NO. 2021rc002) in Guangdong Province, Peoples R China. as well as to the anonymous reviewers for constructive suggestions.

References

Bianco, C.G.L. and Raineri, M. (2017), "An automatic system for the avoidance of wrist singularities in anthropomorphic manipulators", In: *IEEE Conference on Automation Science and Engineering* (*CASE*), Xi'an, China, pp.1302-1309.

Braz-Cesar, M.T. and Barros, R. (2018), "Semi-active fuzzy based control system for vibration reduction of a SDOF structure under seismic excitation", *Smart Struct. Syst.*, *Int. J.*, **21**(4), 389-395. https://doi.org/10.12989/sss.2018.21.4.389

Chen, J., Wang, Q., Cheng, H.H., Peng, W. and Xu, W. (2022a), "A review of vision-based traffic semantic understanding in ITSs", *IEEE Transact. Intell. Transport. Syst.*, **23**(11), 19954-19979. https://doi.org/10.1109/TITS.2022.3182410

Chen, J., Wang, Q., Peng, W., Xu, H., Li, X. and Xu, W. (2022b), "Disparity-based multiscale fusion network for transportation detection", *IEEE Transact. Intell. Transport. Syst.*, **23**(10), 18855-18863. https://doi.org/10.1109/TITS.2022.3161977

Chen, J., Xu, M., Xu, W., Li, D., Peng, W. and Xu, H. (2023a), "A flow feedback traffic prediction based on visual quantified features", *IEEE Transact. Intell. Transport. Syst.*, **24**(9), 10067-10075. https://doi.org/10.1109/TITS.2023.3269794

Chen, Z., Liang, Q., Wei, Z., Chen, X., Shi, Q., Yu, Z. and Sun, T. (2023b), "An overview of in vitro biological neural networks for robot intelligence", *Cyborg Bionic Syst.*, **4**. https://doi.org/10.34133/cbsystems.0001

Cheng, F.T., Hour, T.L., Sun, Y.Y. and Chen, T.H. (1997), "Study and resolution of singularities for a 6-DOF PUMA manipulator", *IEEE Transactions on Systems*, *Man*, *and Cybernetics*, *Part B* (*Cybernetics*), **27**, 332-343. https://doi.org/10.1109/3477.558842

Ding, Y., Zhang, W., Zhou, X., Liao, Q., Luo, Q. and Ni, L.M. (2021), "FraudTrip: Taxi fraudulent trip detection from corresponding trajectories", *IEEE Internet Things J.*, **8**(16), 12505-12517. https://doi.org/10.1109/JIOT.2020.3019398

Fang, H., Ma, S., Wang, J., Zhao, L., Nie, F., Ma, X., Lü, W., Yan, S. and Zheng, L. (2024), "Multimodal In-Sensor Computing Implemented by Easily-Fabricated Oxide-Heterojunction Optoelectronic Synapses", *Adv. Funct. Mater.*, 2409045. https://doi.org/10.1002/adfm.202409045

Fu, Y., Li, C., Yu, F.R., Luan, T.H. and Zhao, P. (2023), "An incentive mechanism of incorporating supervision game for federated learning in autonomous driving", *IEEE Transact. Intell. Transport. Syst.*, **24**(12), 14800-14812. https://doi.org/10.1109/TITS.2023.3297996

Gao, Q., Deng, Z., Ju, Z. and Zhang, T. (2023), "Dual-Hand Motion Capture by Using Biological Inspiration for Bionic Bimanual Robot Teleoperation", *Cyborg Bionic Syst.*, **4**, 52. https://doi.org/10.34133/cbsystems.0052

Gao, J., Tian, S., Yuan, C., Ma, Z., Gao, C., Yan, G., Li, R., Tan, Q. and Zhang, L. (2024), "Design and optimization of a novel double-layer Helmholtz coil for wirelessly powering a capsule robot", *IEEE Transact. Power Electron.*, **39**(1), 1826-1839. https://doi.org/10.1109/TPEL.2023.3321845

Gu, X. and Ren, H. (2023), "A survey of transoral robotic mechanisms: Distal dexterity, variable stiffness, and triangulation", *Cyborg Bionic Syst.*, **4**. https://doi.org/10.34133/cbsystems.0007

Gu, Y., Hu, Z., Zhao, Y., Liao, J. and Zhang, W. (2024), "MFGTN: A multi-modal fast gated transformer for identifying single trawl marine fishing vessel", *Ocean Eng.*, **303**, 117711. https://doi.org/10.1016/j.oceaneng.2024.117711

He, H., Chen, Z., Liu, H., Liu, X., Guo, Y. and Li, J. (2023), "Practical tracking method based on best buddies similarity", *Cyborg Bionic Syst.*, **4**, 50. https://doi.org/10.34133/cbsystems.0050

He, S., Chen, W., Wang, K., Luo, H., Wang, F., Jiang, W. and

Ding, H. (2024a), "Region generation and assessment network for occluded person re-identification", *IEEE Transact. Inform. Forens. Security*, **19**, 120-132. https://doi.org/10.1109/TIFS.2023.3318956

He, S., Luo, H., Jiang, W., Jiang, X. and Ding, H. (2024b), "VGSG: Vision-Guided Semantic-Group Network for Text-Based Person Search", *IEEE Transact. Image Process.*, **33**, 163-176. https://doi.org/10.1109/TIP.2023.3337653

Hou, X., Zhang, L., Su, Y., Gao, G., Liu, Y., Na, Z., Xu, Q., Ding, T., Xiao, L., Li, L. and Chen, T. (2023), "A space crawling robotic bio-paw (SCRBP) enabled by triboelectric sensors for surface identification", *Nano Energy*, **105**, 108013. https://doi.org/10.1016/j.nanoen.2022.108013

Jin, W., Tian, X., Shi, B., Zhao, B., Duan, H. and Wu, H. (2024), "Enhanced UAV Pursuit-Evasion Using Boids Modelling: A Synergistic Integration of Bird Swarm Intelligence and DRL", *Comput. Mater. Continua*, **80**(3), 3523-3553. https://doi.org/10.32604/cmc.2024.055125

Li, S., Chen, J., Peng, W., Shi, X. and Bu, W. (2023), "A vehicle detection method based on disparity segmentation", *Multimedia Tools Applicat.*, **82**(13), 19643-19655. https://doi.org/10.1007/s11042-023-14360-x

Li, Z., Wang, Y., Zhang, R., Ding, F., Wei, C. and Lu, J.G. (2024), "A LiDAR-OpenStreetMap Matching Method for Vehicle Global Position Initialization Based on Boundary Directional Feature Extraction", *IEEE Transact. Intell. Vehicles*. https://doi.org/10.1109/TIV.2024.3393229

Liu, L., Zhang, S., Zhang, L., Pan, G. and Yu, J. (2023), "Multi-UUV Maneuvering Counter-Game for Dynamic Target Scenario Based on Fractional-Order Recurrent Neural Network", *IEEE Transact. Cybernet.*, **53**(6), 4015-4028. https://doi.org/10.1109/TCYB.2022.3225106

Long, S., Huang, W., Wang, J., Liu, J., Gu, Y. and Wang, Z. (2024), "A Fixed-Time Consensus Control With Prescribed Performance for Multi-Agent Systems Under Full-State Constraints", *IEEE Transact. Automat. Sci. Eng.*, 1-10. https://doi.org/10.1109/TASE.2024.3445135

Mi, C., Liu, Y., Zhang, Y., Wang, J., Feng, Y. and Zhang, Z. (2023), "A vision-based displacement measurement system for foundation pit", *IEEE Transact. Instrum. Measur.*, **72**. https://doi.org/10.1109/TIM.2023.3311069

Sheng, H., Wang, S., Chen, H., Yang, D., Huang, Y., Shen, J. and Ke, W. (2024), "Discriminative feature learning with co-occurrence attention network for vehicle ReID", *IEEE Transact. Circuits Syst. Video Technol.*, **34**(5), 3510-3522. https://doi.org/10.1109/TCSVT.2023.3326375

Shi, Y., Hou, X., Na, Z., Zhou, J., Yu, N., Liu, S., Xin, L., Gao, G. and Liu, Y. (2023), "Bio-inspired attachment mechanism of dynastes Hercules: Vertical climbing for on-orbit assembly legged robots", *J. Bionic Eng.*, **21**(1), 137-148. https://doi.org/10.1007/s42235-023-00423-0

Song, F., Liu, Y., Dong, Y., Chen, X. and Tan, J. (2024), "Motion Control of Wafer Scanners in Lithography Systems: From Setpoint Generation to Multi-Stage Coordination", *IEEE Transact. Instrum. Measur.* https://doi.org/10.1109/TIM.2024.3413202

Sun, R., Dai, Y. and Cheng, Q. (2023), "An Adaptive Weighting Strategy for Multisensor Integrated Navigation in Urban Areas", *IEEE Internet Things J.*, **10**(14), 12777-12786. https://doi.org/10.1109/JIOT.2023.3256008

Sun, J., Zhou, L., Geng, B., Zhang, Y. and Li, Y. (2024a), "Leg State Estimation for Quadruped Robot by Using Probabilistic Model With Proprioceptive Feedback", *IEEE/ASME Transact. Mechatron.*, 1-12. https://doi.org/10.1109/TMECH.2024.3421251

Sun, Y., Peng, Z., Hu, J. and Ghosh, B.K. (2024b), "Event-triggered critic learning impedance control of lower limb exoskeleton robots in interactive environments", *Neurocomputing*, **564**, 126963. https://doi.org/10.1016/j.neucom.2023.126963

Tian, H., Wang, X., Zhou, G., Nam, K. and Ji, Z. (2023), "A criterion of robustness intelligent nonlinear control for multiple time-delay systems based on fuzzy Lyapunov methods", *Nonlin. Dyn.*, **76**, 23-31. https://doi.org/10.1007/s11071-013-0869-9

Tian, H., Zhao, M., Liu, J., Wang, Q., Yu, X. and Wang, Z. (2024), "Dynamic Analysis and Sliding Mode Synchronization Control of Chaotic Systems with Conditional Symmetric Fractional-Order Memristors", *Fract. Fraction.*, **8**(6), 307. https://doi.org/10.3390/fractalfract8060307

Wang, B., Zheng, W., Wang, R., Lu, S., Yin, L., Wang, L., Yin, Z. and Chen, X. (2024a), "Stacked Noise Reduction Auto Encoder–OCEAN: A Novel Personalized Recommendation Model Enhanced", *Systems*, **12**(6), 188. https://doi.org/10.3390/systems12060188

Wang, J., Wu, Y., Chen, C.L.P., Liu, Z. and Wu, W. (2024b), "Adaptive PI event-triggered control for MIMO nonlinear systems with input delay", *Inform. Sci.*, **677**, 120817. https://doi.org/10.1016/j.ins.2024.120817

Wang, X., Zhang, R., Miao, Y., Wang, S. and Zhang, Y. (2024c), "PI^2-Based Adaptive Impedance Control for Gait Adaption of Lower Limb Exoskeleton", *IEEE/ASME Transact. Mechatron.* https://doi.org/10.1109/TMECH.2024.3370954

Wang, Y., Sun, R., Cheng, Q. and Ochieng, W.Y. (2024d), "Measurement quality control aided multisensor system for improved vehicle navigation in urban areas", *IEEE Transact. Industr. Electron.*, **71**(6), 6407-6417. https://doi.org/10.1109/TIE.2023.3288188

Wang, Z., Chen, M., Guo, Y., Li, Z. and Yu, Q. (2024e), "Bridging the Domain Gap in Satellite Pose Estimation: A Self-Training Approach Based on Geometrical Constraints", *IEEE Transact. Aerosp. Electron. Syst.*, **60**(3), 2500-2514. https://doi.org/10.1109/TAES.2023.3250385

Wang, Z., Gao, D., Deng, K., Lu, Y., Ma, S. and Zhao, J. (2024f), "Robot base position and spacecraft cabin angle optimization via homogeneous stiffness domain index with nonlinear stiffness characteristics", *Robot. Comput.-Integr. Manuf.*, **90**, 102793. https://doi.org/10.1016/j.rcim.2024.102793

Xu, B. and Guo, Y. (2022), "A novel DVL calibration method based on robust invariant extended Kalman filter", *IEEE Transact. Vehicular Technol.*, **71**(9), 9422-9434. https://doi.org/10.1109/TVT.2022.3182017

Xu, B., Wang, X., Zhang, J., Guo, Y. and Razzaqi, A.A. (2022), "A novel adaptive filtering for cooperative localization under compass failure and non-gaussian noise", *IEEE Transact. Vehicular Technol.*, **71**(4), 3737-3749. https://doi.org/10.1109/TVT.2022.3145095

Xu, X., Fu, X., Zhao, H., Liu, M., Xu, A. and Ma, Y. (2023), "Three-Dimensional Reconstruction and Geometric Morphology Analysis of Lunar Small Craters within the Patrol Range of the Yutu-2 Rover", *Remote Sensing*, **15**(17), 4251. https://doi.org/10.3390/rs15174251

Yang, M., Cai, C., Wang, D., Wu, Q., Liu, Z. and Wang, Y. (2024), "Symmetric differential demodulation-based heterodyne laser interferometry used for wide frequency-band vibration calibration", *IEEE Transact. Industr. Electron.*, **71**(7), 8132-8140. https://doi.org/10.1109/TIE.2023.3299015

Yin, L., Wang, L., Lu, S., Wang, R., Ren, H., AlSanad, A., AlQahtani, S.A., Yin, Z., Li, X. and Zheng, W. (2024a), "AFBNet: A Lightweight Adaptive Feature Fusion Module for Super-Resolution Algorithms", *Comput. Model. Eng. Sci.*, **140**(3), 2315-2347. https://doi.org/10.32604/cmes.2024.050853

Yin, Y., Wang, Z., Zheng, L., Su, Q. and Guo, Y. (2024b), "Autonomous UAV Navigation with Adaptive Control Based on Deep Reinforcement Learning, *Electronics*, **13**(13), 2432.

https://doi.org/10.3390/electronics13132432
Zhang, J., Yang, D., Li, W., Zhang, H., Li, G. and Gu, P. (2024), "Resilient output control of multiagent systems with DoS attacks and actuator faults: Fully distributed event-triggered approach", *IEEE Transact*. *Cybernet*., 1-10. https://doi.org/10.1109/TCYB.2024.3404010
Zheng, C., An, Y., Wang, Z., Wu, H., Qin, X., Eynard, B. and Zhang, Y. (2022), "Hybrid offline programming method for robotic welding systems", *Robot*. *Comput*.-*Integr*. *Manuf*., **73**, 102238. https://doi.org/10.1016/j.rcim.2021.102238
Zheng, C., An, Y., Wang, Z., Qin, X., Eynard, B., Bricogne, M., Le Duigou, J. and Zhang, Y. (2023), "Knowledge-based engineering approach for defining robotic manufacturing system architectures", *Int*. *J*. *Product*. *Res*., **61**(5), 1436-1454. https://doi.org/10.1080/00207543.2022.2037025
Zhou, H., Cao, S., Zhang, S., Li, F. and Ma, N. (2023a), "Design of a fuel explosion-based chameleon-like soft robot aided by the comprehensive dynamic model", *Cyborg Bionic Syst*., 4. https://doi.org/10.34133/cbsystems.0010
Zhou, Z., Wang, Y., Zhou, G., Nam, K., Ji, Z. and Yin, C. (2023b), "A Twisted Gaussian Risk Model Considering Target Vehicle Longitudinal-Lateral Motion States for Host Vehicle Trajectory Planning", *IEEE Transact*. *Intell*. *Transport*. *Syst*., **24**(12), 13685-13697. https://doi.org/10.1109/TITS.2023.3298110
Zhou, Y., Xie, J., Zhang, X., Wu, W. and Kwong, S. (2024), "Energy-Efficient and Interpretable Multisensor Human Activity Recognition via Deep Fused Lasso Net", *IEEE Transact*. *Emerg*. *Topics Computat*. *Intell*., 1-13. https://doi.org/10.1109/TETCI.2024.3430008
Zhu, C. (2023), "Intelligent robot path planning and navigation based on reinforcement learning and adaptive control", *J*. *Logist*. *Inform*. *Service Sci*., **10**(3), 235-248. https://doi.org/10.33168/JLISS.2023.0318