Available online at www.sciencedirect.com
ScienceDirect
August 2017, 24(4): 51–56
www.sciencedirect.com/science/journal/10058885
The Journal of China Universities of Posts and Telecommunications
http://jcupt.bupt.edu.cn

Autonomous parking control for intelligent vehicles based on a novel algorithm

Gao Hongbo[1], Xie Guotao[1], Zhang Xinyu[2] (✉), Cheng Bo[1]

1. State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing 100084, China
2. Information Technology Center, Tsinghua University, Beijing 100084, China

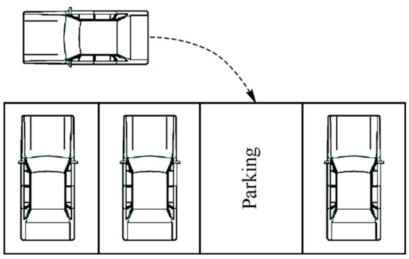
Abstract

Along with the increasing number of vehicles, parking space becomes narrow gradually, safety parking puts forward higher requirements on the driver's driving technology. How to safely, quickly and accurately park the vehiclo to parking space right? This paper presents an automatic parking scheme based on trajectory planning, which analyzing the mechanical model of the vehicle, establishing vehicle steering model and parking model, coming to the conclusion that it is the turning radius is independent of the vehicle speed at low speed. The Matlab simulation environment verifies the correctness and effectiveness of the proposed algorithm for parking. A class of the automatic parking problem of intelligent vehicles is solved.

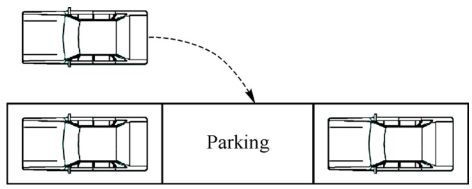
Keywords intelligent vehicles, control algorithm, autonomous parking

1 Introduction

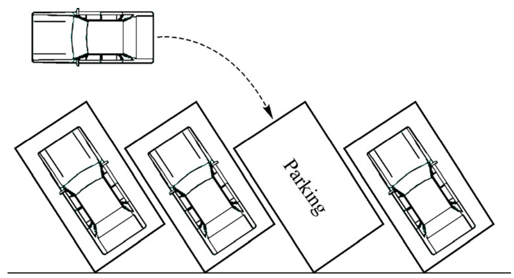
With the development of industrial automation, the number of cars is increasing, parking space is more and more nervous, and parking is a specific problem. Therefore, people put forward higher requirements for convenient operation for a vehicle, hope the vehicle can automatically drive parking [1–2]. City parking spaces are divided into three kinds, which as shown in Fig. 1.



(a) Vertical parking



(b) Parallel parking



(c) Inclined parking

Fig. 1 Parking space

Automatic parking system can solve the above problems, it uses sensor technology, computer technology and automatic control technology to accurately perceive the parking environment, at the same time to plan an optimal and feasible parking path, and then control the vehicle to

Received date: 19-12-2016
Corresponding author: Zhang Xinyu, E-mail: xyzhang@tsinghua.edu.cn
DOI: 10.1016/S1005-8885(17)60223-1

accomplish trajectory tracking, and then parking the vehicle in parking spaces accurately and safely.

The research and development of automatic parking system are early at abroad, the domestic is still in the initial stage. The technical scheme of the automatic parking driving system is complex, high cost, it is widely used in high-end models [3–4]. The research of automatic parking algorithm is a path planning method, the other is the control algorithm based on the experience. The path planning method is planning a suitable geometric parking path, and then consider the dynamics and kinematics model of environmental constraints, adjust geometric parameters of the parking path then generate parking trajectory. The algorithm relies on the sensing accuracy of the surrounding environment and the control precision of the controller [5–6].

Lyon [7] proposed a method of fitting a 5° polynomial curve through environmental constraints to complete the guidance of the car's parking trajectory. Kanayama et al. [8] used the sensor as the environment perception obtained the parking environment and the position information of the vehicle, generated a feasible parking trajectory which conforms to the environmental restriction. Hsieh et al. [9] proposed the automatic parking in the limit space, which includes two aspects of the control of speed and steering angle. Chen et al. [10] designed a hierarchical adaptive method for the automatic parking by the multi-loop control algorithm accomplish trajectory tracking control. The high accuracy laser radar scans the environment to obtain the parking positioning, speed sensor obtains the position of the vehicle, and the hierarchical adaptive method computes three stages parking path.

The control algorithm bases on the experience by fuzzy control or artificial intelligence methods such as neural network learn the driver's parking experience, simulates human driving behavior, accomplish to real-time automatic parking. Gómez-Bravo et al. [11] proposed the fuzzy control method to achieve parallel parking, designed the fuzzy controller for the limited environment of this special scene. Li et al. [12] used an evolutionary algorithm to design the fuzzy controller to realize automatic parking, complete automatic parking. Paromtchik et al. [13] proposed the nonlinear feedforward feedback and linear feedback automatic parking method combined with artificial neural network technology, analyzed the stability of the automatic parking controller and optimize of the parking path. Wang et al. used neural network to accomplish automatic parking control, and used fuzzy self-organizing neural network combined with the online learning algorithm to accomplish the network parameters optimized into reduce control error, improved the accuracy of real-time control [14–15].

In this paper, we mainly concentrate on vertical parking problems. A novel autonomous parking control algorithm is proposed. We use a circular arc to design a reference path which is easy for a vehicle to follow and also computationally efficient. In the simulation, this algorithm is efficient with different initial positions. Actually, we are going to make it implemented in intelligent vehicles in future work. Therefore, we considered this algorithm some restrictions that a vehicle may have such as mechanical restrictions, sensor type restrictions, and etc.

2 Parking control algorithm

2.1 Kinematic model

As shown in Fig. 2, it is the Ackerman steering model for the vehicle. (x, y) represent the coordinates of the center point of the vehicle rear axle, θ represents the yaw angle of vehicle.

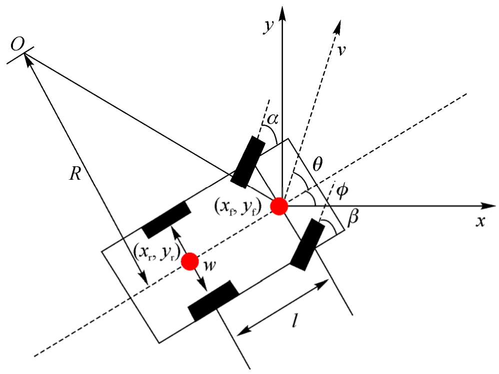


Fig. 2 Ackerman vehicle model

As shown in Fig. 2, R is the turning radius of the vehicle. l is wheelbase, w is the track, α and β are the left and right wheel steering angle. The equations of motion for this model can be written as

$$\cot\beta - \cot\alpha = \frac{w}{l} \tag{1}$$

$$\cot\theta = \cot\alpha + \frac{w}{2l} \tag{2}$$

$$\cot\theta = \cot\beta - \frac{w}{2l} \tag{3}$$

$$\cot\theta = \frac{\cot\alpha + \cot\beta}{2} \tag{4}$$

$$R = \frac{l}{\tan\theta} \tag{5}$$

where θ is the front wheel steering angle of vehicle, the point O is the turning center of vehicle, v is the speed of the vehicle, the direction of the tangent direction of vehicle turning, (x_f, y_f) and (x_r, y_r) are the front and rear axle center coordinates of vehicle respectively. (x_{lr}, y_{lr}) is the left rear wheel coordinates, (x_{rr}, y_{rr}) is the coordinates of the right rear wheel. The vehicle does not occur sideslip phenomenon in the process of parking, the constraint equation of the rear wheel can be written as

$$y_r'\cos\phi - x_r'\sin\phi = 0 \tag{6}$$

The front axle center and rear axle position constraint can be written as

$$\left.\begin{aligned} x_f &= x_r + l\cos\phi \\ y_f &= y_r + l\sin\phi \end{aligned}\right\} \tag{7}$$

The derivation of Eq. (7), the center point of the front axle speed can be written as

$$\left.\begin{aligned} x_f' &= x_r' + \phi' l\sin\phi \\ y_f' &= y_r' + \phi' l\cos\phi \end{aligned}\right\} \tag{8}$$

Combining the Eqs. (6)–(8), simultaneous solution can be written as

$$x_f'\sin\phi - y_f'\cos\phi + l\phi' = 0 \tag{9}$$

The decomposed velocity of front wheel center point can be written as

$$\left.\begin{aligned} x_f' &= v\cos(\theta+\phi) \\ y_f' &= v\sin(\theta+\phi) \end{aligned}\right\} \tag{10}$$

Substituting Eq. (5) into Eq. (4) and then obtains yaw rate can be written as

$$\left.\begin{aligned} \phi' &= -v\frac{\sin\theta}{l} \\ \phi &= \int -v\frac{\sin\theta}{l}\mathrm{d}t = v\frac{\cos\theta}{l} \end{aligned}\right\} \tag{11}$$

The rear axle center point speed can be written as

$$\left.\begin{aligned} x_r' &= v\cos\theta\cos\phi \\ y_r' &= v\sin\theta\cos\phi \end{aligned}\right\} \tag{12}$$

Integral Eqs. (5) and (12), then obtains trajectory equation of the center of the rear wheel

$$(x_r - l\sin\phi\cos\theta)^2 + (y_r - l\cos\phi\cos\theta)^2 = (l\cot\theta)^2 \tag{13}$$

As can be seen from the Eq. (13), when the Ackerman steering model, the trajectory of the vehicle is composed of a number of circular arcs curve.

2.2 Calculate the parking point

The target of automatic parking is parked the vehicle into the designated parking spaces, the center point of parking entrance represent end of the parking, the coordinate of P represents in the geodetic coordinate system, when the parking point is certain, parking algorithm will calculate a circular arc makes the starting point and end point in the parking arc. And then the parking controller control the front wheel steering angle following the planned trajectory so that the vehicle can quickly and safely reach the designated parking. When the vehicle reaches the parking, the direction of the vehicle is parallel to the parking space, the parking controller will switch the driving mode from the circular arc to a straight line, the vehicle will stop in the middle of the parking space. The parking trajectory as shown in Fig. 3, the y axis of the coordinate is the forward direction of the vehicle, the carrier coordinate is moving with the vehicle, parking point is a dynamic point in the carrier coordinates, which changes with the vehicle.

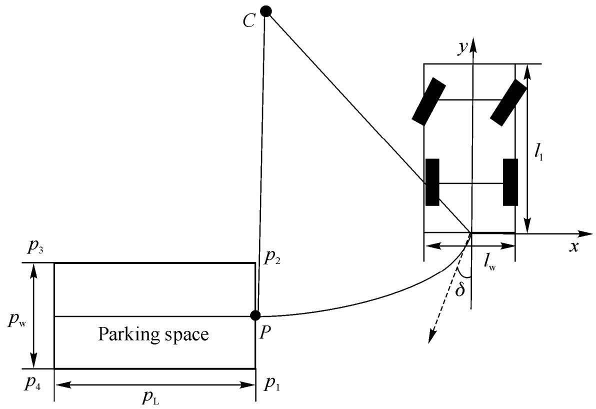


Fig. 3 Parking trajectory planning

In each control cycle, the coordinate position of the parking spot is calculated in real time according to the current position and direction of the car, and the position of the parking spot can be written

$$x_P = \frac{x_{p_1} + x_{p_2}}{2} \tag{14}$$

$$y_P = \frac{y_{p_1} + y_{p_2}}{2} \tag{15}$$

where (x_{p_1}, y_{p_1}) is the coordinate of p_1, (x_{p_2}, y_{p_2}) is the coordinate of p_2, and (x_P, y_P) is the coordinate of P.

As shown in Fig. 3, the four points of p_1, p_2, p_3 and p_4 are vertex of vertices parking spaces, C is the center of circular trajectory planning. l_l and l_w are the length and width of the vehicle, p_L and p_w are the long and wide parking spaces, and the point P is the end parking point.

2.3 Parking control algorithm

The process of automatic parking algorithm as shown in Fig. 4, the vehicle is the parking state, it gets the target parking and vehicle coordinate information by the sensors information, using an arc to connect the two points, and calculated the radius of R and C.

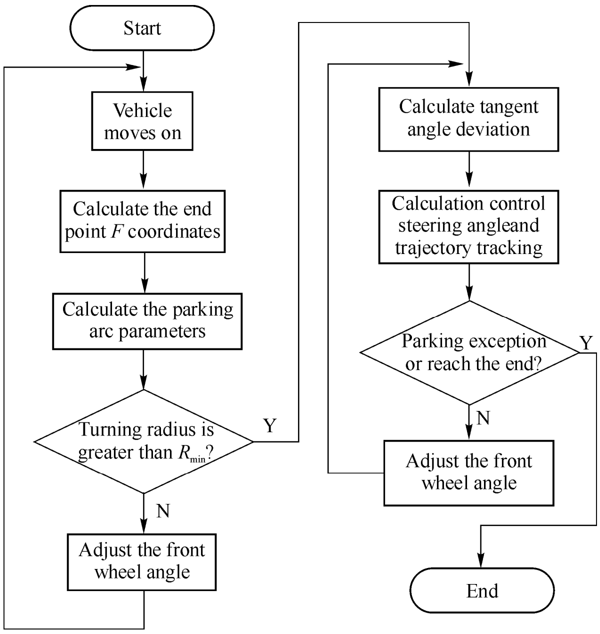


Fig. 4 Automatic parking algorithm flow chart

Comparing the turning radius to R_{min}, when $R > R_{min}$, then go into parking. When $R \leqslant R_{min}$, the parking radius is too small, the vehicle cannot adjust the steering wheel to achieve parking trajectory tracking. The vehicle must be driving the forward or backward in a distance, then get the target parking and vehicle coordinate information by the sensors information, using an arc to connect the two points, and calculated the radius of R and the coordinate of C again. If it tries several times cannot find a suitable parking trajectory, then stop the automatic parking. When the track radius is less than R_{min} of the arc or the combination of straight line, the vehicle cannot be done by turning in the same place to complete the parking task. As shown in Fig. 3, the y axis of vehicle is tangent to the arc trajectory in the parking trajectory and is parking trajectory tracking object, it is the center of C and x axis, it can be written as

$$\delta = \arctan \frac{y_c}{x_c} = 0 \tag{16}$$

By adjusting the desired tracking angle , in order to reduce the wear of the tire and the tracking error, reduce the overshoot of the trajectory tracking, design the increment proportional integral derivative (PID) controller, in order to reduce the overshoot, the integral separation and integral limiting method are used for the integral, the incomplete differential method is used for the differential. The increment PID controller can be written as

$$u(k-1) = K_P \left(e(k-1) + \frac{T}{T_I} \sum_{i=0}^{k-1} e(i) + \frac{T_D}{T} (e(k-1) - e(k-2)) \right) \tag{17}$$

where K_P is proportional coefficient, $u(\cdot)$ is the control variable, T is the sampling cycle, T_I is the integral time constant, T_D is the differential time constant, $e(\cdot)$ is the error between the previous cycle and the next cycle.

The relationship between the former control cycle and the next control cycle can be written as

$$u(k) = u(k-1) + \Delta u(k) \tag{18}$$

Control increment can be written as

$$\Delta u(k) = K_P \left((e(k) - e(k-1)) + \frac{T}{T_I} e(k) + \frac{T_D}{T} (e(k) - 2e(k-1) + e(k-2)) \right) \tag{19}$$

When the time constant of the system and the deviation is larger, it will have a greater integral value. When the integral value is too large, it will make the controller output saturation. When the deviation is eliminated, the output of the controller is still in a saturated state due to the larger integral value. When it has the control output, this will produce a large overshoot. When the deviation is large, the control of the integral action is separated. When the deviation is small, and the integral action is added. As shown in Eq. (20).

$$u(k) = \begin{cases} K_P \left(e(t) + \dfrac{1}{T_I} \displaystyle\int_0^t e(t)\mathrm{d}t + T_D \dfrac{\mathrm{d}e(t)}{\mathrm{d}t} \right); & |e(k)| \leqslant \varepsilon \\ K_P \left(e(t) + T_D \dfrac{\mathrm{d}e(t)}{\mathrm{d}t} \right); & |e(k)| \geqslant \varepsilon \end{cases} \tag{20}$$

where ε is an integral separate threshold.

When the deviation is greater than ε, it removes the integral step, the system adopts proportional derivative (PD) control. When the system deviation is less than ε, it adds the integral step, the system use the PID control at this time. The integral separation control method can reduce the overshoot of the system and reduce the static error of the system.

The differential is adding a first-order inertia link before in the front of the differential link, generating incomplete differentiation, as shown in Fig. 5. In Fig. 5, T_f is time coefficient of complex variable, s is the complex variable.

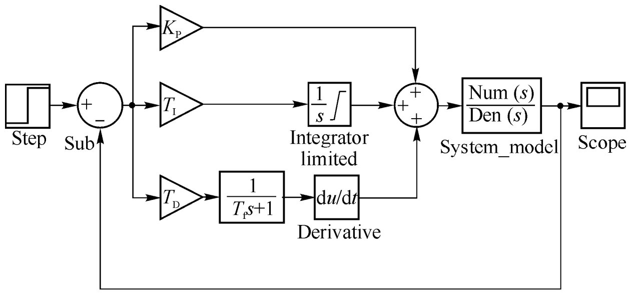


Fig. 5 System of incomplete differential PID

3 Simulations and results

3.1 Parking trajectory planning

Construct simulation environment in Matlab and Simulink, call the interface function in Simulink simulation environment to obtain the information of parking spaces and the location information of the intelligent vehicle.

The end of the parking is certain, then call the function of IVehicle.Point_F to obtain the available parking spaces location. IVehicle.Point_F defines a coordinate array of available parking space.

Parking trajectory center is certain, then call the function of state_estimate.unpack(·) to obtain the autonomous vehicle of position. Use the end parking point and tangent line in the parking, parking trajectory radius R can be written as

$$R = 2\left|\frac{(x_A - x_P)^2 + (y_A - y_P)^2}{x_A - x_P}\right| \tag{21}$$

where the coordinate of (x_A, y_A) is the position of the intelligent vehicle.

The intelligent vehicle drive into parking space from left, parking trajectory center coordinates can be written as

$$\left.\begin{aligned} x_c &= x_P - R \\ y_c &= y_P \end{aligned}\right\} \tag{22}$$

The intelligent vehicle drive into parking space from right, parking trajectory center coordinates can be written as

$$\left.\begin{aligned} x_c &= x_P + R \\ y_c &= y_P \end{aligned}\right\} \tag{23}$$

When vehicle tracks parking trajectory tracking completely, the y axis of the vehicle is along the tangent direction of a circular arc. The y axis of the vehicle is tangent tracking arc that can keep tracking the trajectory planning arc. The y axis of the vehicle and the tangent angle can be written as

$$\delta_{\text{target}} = \arctan\left(-\frac{x_c}{y_c}\right) \tag{24}$$

where the y axis of expectations and the tangent angle is δ_{target}, and $\delta_{\text{target}} = 0$.

3.2 Simulation results and analysis

Fig. 6 is the step response experiment data of the intelligent vehicle steering system. A unit step signal is applied to the steering control system when the intelligent vehicle drives in a straight line. The steering signal data of the intelligent vehicle is recorded, and the curve is shown in Fig. 6. In Fig. 6, the overshoot of the system is 8%, the rise time is 0.3 s, the peak time is 0.41 s, the adjusting time is 0.45 s.

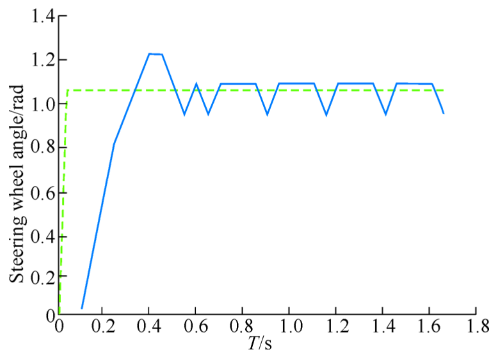


Fig. 6 Step response curve of steering control

As shown in Fig. 7, the intelligent vehicle finds a suitable parking space, the trajectory planning algorithm complete the intelligent vehicle automatic parking. In Fig. 7, the blue box represents the intelligent vehicle, the green box represents the end parking location of the intelligent vehicle. The letter P_1, P and C represents the starting point, the end point and the center point of the parking trajectory respectively. The red circular arc curve is the expectation parking trajectory obtained by the parking trajectory planning algorithm. In Fig. 7, the intelligent vehicle can be track the planning of the parking trajectory completely, and complete automatic parking.

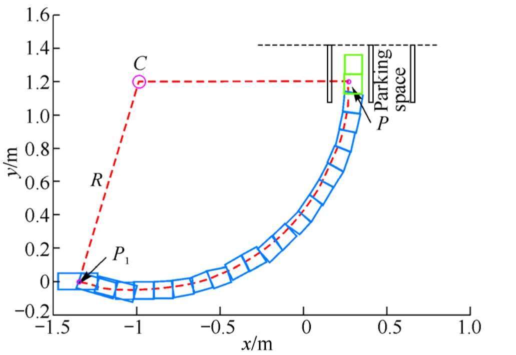


Fig. 7 Intelligent vehicle parking simulation experiment

Fig. 8 shows that the radius of parking is less than the minimum turning radius of the intelligent vehicle, intelligent vehicle straight away from a distance, then calculate the parking trajectory again, until the appropriate parking trajectory can be found and complete automatic parking.

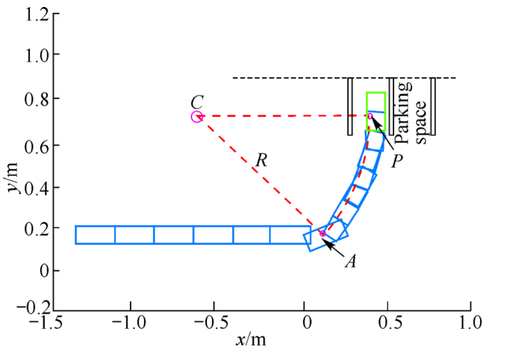


Fig. 8 Search parking spaces and complete the automatic parking process

4 Conclusions

The automatic parking system is a very complicated system, which involves image processing, pattern recognition, motion control technology, and so on. Automatic parking also has broad market demands and prospects, therefore, the realization of automatic parking on the real vehicle is an important direction for the future research and development. Automatic parking control algorithm is proposed in this paper, The Matlab and Simulink simulation environment verify the correctness and effectiveness of the proposed algorithm for parking.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (61035004, 61273213, 61300006, 61305055, 90920305, 61203366, 91420202), the National Hi-Tech Research and Development Program of China (2015AA015401), the National Basic Research Program of China (2016YFB0100906, 2016YFB0100903), the Junior Fellowships for Advanced Innovation Think-Tank Program of China Association for Science and Technology (DXB-ZKQN-2017-035), the Project Funded by China Postdoctoral Science Foundation, and the Beijing Municipal Science and Technology Commission Special Major (D171100005017002).

References

1. Wang W C, Chen R S. A vision-based fuzzy logic controller for backing-up an autonomous vehicle. Journal of Intelligent and Fuzzy System, 2008, 19(4): 273-284
2. Gao H B, Liu Y C, Zhang X Y, et al. Path planning algorithm regarding rapid parking based on static optimization. Proceedings of the 3rd IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS'14), Nov 27-29, 2014, Shenzhen, China. Piscataway, NJ, USA: IEEE, 2014: 672-678
3. Seo Y W, Urmson C, Wettergreen D, et al. Building lane-graphs for autonomous parking. Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'10), Oct 18-22, 2010, Taipei, China. Piscataway, NJ, USA: IEEE, 2010: 6052-6057
4. Gao H B, Zhang X Y, Liu Y C, et al. Cloud model approach for lateral control of intelligent vehicle systems. Scientific Programming, 2016: Article 6842891/1-12
5. Ali M, Rahmat A, Ramin S. Bidding strategy of vehicle parking for participating in energy and spinning reserve markets. Berlin: Springer-Verlag, 2014: 403-411
6. Gao H B, Zhang X Y, An L F, et al. Relay navigation strategy study on intelligent drive on urban roads. Journal of China University of Posts and Telecommunications, 2016, 23(2): 79-90
7. Lyon D. Parallel parking with curvature and nonholonomic constraints. Proceedings of the 1992 IEEE Intelligent Vehicles Symposium, Jun 29-Jul 1, 1992, Detroit, MI, USA. Piscataway, NJ, USA: IEEE, 1992: 341-346
8. Kanayama Y J, Hartman B I. Smooth local path planning for autonomous vehicles. Proceedings of the 1989 International Conference on Robotics and Automation: Vol 3, May 14-19, 1989, Scottsdale, AZ, USA. Piscataway, NJ, USA: IEEE, 1989: 1265-1270
9. Hsieh M F, Ozguner U. A parking algorithm for an autonomous vehicle. Proceedings of the 2008 IEEE Intelligent Vehicle Symposium, Jun 4-6, 2008, Eindhoven, Netherlands. Piscataway, NJ, USA: IEEE, 2008: 1155-1160
10. Chen J J, Mei T, Zhu H. Study of autonomous vehicle parking system based on path planning and the double closed loop method. Electronic Measurement Technology, 2011, 34(3): 20-24 (in Chinese)
11. Gómez-Bravo F, Cuesta F, Ollero A. Parallel and diagonal parking in nonholonomic autonomous vehicles. Engineering Applications of Artificial Intelligence, 2001, 14(4): 419-434
12. Li T H S, Chang S J. Autonomous fuzzy parking control of a car-like mobile robot. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, 2003, 33(4): 451-465
13. Paromtchik I E, Laugier C. Autonomous parallel parking of a nonholonomic vehicle. Proceedings of the 1996 IEEE International Conference on Intelligent Vehicles, Sept 19-20, 1996, Tokyo, Japan. Piscataway, NJ, USA: IEEE, 1996: 13-18
14. Wang D B, Liang H W, Yang N N, et al. On path planning of two self-parking methods. Electronic Measurement Technology, 2011, 34(1): 27-38 (in Chinese)
15. Gao H B, Zhang X Y, Zhang T L, et al. Research of intelligent vehicle variable granularity evaluation based on cloud model. Acta Electronica Sinica, 2016, 44(2): 365-373 (in Chinese)

(Editor: Wang Xuying)