Implementation of an Autonomous Parking System in a Parking Lot

Po-Kai Tseng, Ming-Hung Hung, Po-Kai Yu, Sheng-Wei Chang, and Tong-Wen Wang  
Department of Green Vehicle Development Division, Automotive Research & Testing Center  
No.6, Lugong S. 7th Rd., Lukang, Changhua County 50544, Taiwan (R.O.C.)  
E-mail: {pktseng, hung, kainonex, jasonchang, ericwang55}@artc.org.tw

Abstract

An autonomous parking system (APS) is realized in this paper. In the proposed APS, the user can get out the vehicle at the entrance of the parking lot and uses the tablet to park vehicle automatically. The proposed APS includes a server and multiple cameras. Multiple cameras are equipped around the parking lot to real-time monitor parking spaces. Once a parking space is detected, the image of the parking space is sent to the server. The server processes the image, marks the unoccupied parking space on the map of the parking lot, and sends the marked map to the user’s tablet via wireless communication, such as 3G/Wifi. An interface is developed and installed in user’s tablet to receive the marked map from the server and to display which parking spaces are unoccupied on the map. After user selects an unoccupied parking space on the tablet, a command is sent to user’s vehicle. The vehicle then goes to the booked parking space and performs the back-in parking automatically. A path tracking scheme is proposed to navigate the vehicle. The PT scheme adopts the concepts of pure pursuit algorithm and kinematic steering condition to obtain steering angle. Implementation results show that the proposed APS can successfully guide vehicle to park in a parking lot automatically.

Keyword: autonomous parking system, back-in parking, pure pursuit, kinematic steering condition.

Introduction

Vehicle is one of the most used transportations in the modern city. To reduce vehicle accidents, many advanced assistant safety techniques, such as lane departure warning (LDW) [1] system, forward collision warning (FCW) [2] system, and blind spot warning (BSW) system [3], have been developed on the vehicle. An assistant safety system has become indispensable for
up-to-date vehicle. With the increase of vehicles, available parking spaces have become more and more scarce. Due to the reduction of the parking space, potential accidents are increased in the duration of parking. To reduce the accidents, a parking assistant system (PAS) is presented in [4]. The PAS uses the rear camera and ultrasonic sensor to detect rear obstacle to assist driver for parking. To improve PAS, the advanced parking guidance system (APGS) [5, 6] is presented. In APGS, the surrounding environment is sensed through camera and ultrasonic sensors to find suitable parking space. When a suitable parking space is found, a reversing trajectory is calculated for parking on the found parking space.

In this paper, an autonomous parking system (APS) is realized in a parking lot. In our proposed APS, parking spaces in the parking lot are detected by cameras. A server is set up to create a parking lot map and the driving path. Once a parking space is detected from camera, a message is sent to server for the update of the parking lot map. When a vehicle enters the parking lot, driver only needs to stop it at the entrance of the parking lot. Driver gets out of the vehicle and uses a tablet to perform parking. Driver requests server through tablet to obtain the parking lot map and driving path, and selects a parking space for parking. Once a parking space is selected, a message generated by tablet is sent to server for the update of the parking lot map and another message generated by tablet is sent to vehicle to trigger autonomous parking.

Once the vehicle receives the message from tablet, the vehicle starts the autonomous parking. A path tracking (PT) scheme is proposed to navigate the vehicle to track the driving path. The PT scheme uses the GPS information as the input and adopts the concepts of pure pursuit algorithm and kinematic steering condition to derive steering angle. A CAN-based transmission interface is built to assist the communications in the vehicle. To verify the proposed APS, APS is implemented in a parking lot. Implementation results indicate that the APS can successfully guide the vehicle to the selected parking space.

**System Architecture**

The architecture of our APS is shown in Fig. 1. The proposed APS is composed of cameras, server, tablet, and vehicle. Initially, all available parking spaces and parking lot environment are built up as a parking lot map in a server in advance. Cameras are set up around parking lot to monitor parking spaces.

Once a camera detects that a parking space is unoccupied, this information is sent to server to update parking lot map. Server is responsible to maintain parking lot map. When a vehicle enters this parking lot, user get out of the vehicle and uses the tablet to request server to provide the parking lot map and driving path. User selects an unoccupied parking space and
the tablet then sends the selected parking space to vehicle and server. When server received the information from tablet, the server updates the parking lot map. The vehicle starts to track driving path and perform back-in parking when the vehicle reaches the selected parking space.

All equipments of the experimental vehicle are shown in Fig. 2. A wheel speed sensor (WSS) is installed on the vehicle to count the traveling distance. One real time kinematic GPS (RTK-GPS) is roof-mounted on the center of rear axle to provide the real-time GPS position and the heading of the vehicle. The data rate of RTK-GPS is at 20Hz.

A microcontroller unit (MCU) acts the main decision maker, where signals of steering wheel, gear, ultrasonic sensor, PC1, and PC2 are delivered on the controller area network (CAN) via the message oriented transmission protocol [7, 8]. Each signal is less than 8 bytes with using standard ID. Since the speed of the vehicle is set as idle speed, we do not control the throttle. The PC1 is to deal with the information from RTK-GPS. The RTK-GPS data follows the NMEA [9] form. A C# program is used to get the latitude, longitude, and heading from $GPRMC. The $GPRMC of NMEA is shown in Fig. 3. The PC2 is a listening client to wait a message from tablet and then command to MCU.

An electric power steering (EPS) system [5] is installed on the steering wheel of the vehicle. The EPS includes, motor, steering angle sensor, and power width modulation (PWM) to control steering wheel and sense the steering angle.

All the wireless communications between camera, server, tablet, and vehicle follows the IEEE 802.11b standard. The database of server is built by Microsoft SQL Server.
Driving Path Generation and Parking Space Recognition

An experimental parking lot is selected in ARTC at Taiwan. The experimental parking lot composes of 16 parking spaces, which is shown in Fig. 4(a). Cameras are set up around the parking lot to monitor parking spaces. We generate the driving path for the parking vehicle in advance such that the parking vehicle can track the driving path to reach the booked parking space. In Fig. 4(a), 6 reference locations (i.e., the dashed arrows) are measured manually by RTK-GPS. Based on the 6 reference points, the entrance point, exit point, and four corners (i.e., the solid arrows), are calculated via interpolation method. Figure 4(b) shows the corresponding RTK-GPS position on the google map. The driving path (i.e., dotted arrows) then can be derived via interpolation method.

The location and view of each camera is shown in Fig. 5(a). The parking space is recognized by analyzing the value of entropy of image for the region of interest. The region of interest is set as each parking space. An entropy threshold is set to determine whether a parking space is occupied. If the entropy of a parking space exceeds the threshold, we regards the parking
space is occupied. Two implementations of parking space recognition are shown in Fig. 5(b), where the five parking spaces are mapping to the plaid at the upper right corner. The second and third plaids become red when a vehicle occupies the corresponding parking spaces. Once a parking space is detected due to occupation, this information is sent to server for the maintenance of parking lot map.

(a) The location and view of each camera  
(b) parking space recognition

Figure 5. Implementation of parking space recognition

Parking Path Generation

The parking path is generated by the previous proposed work, automatic parking system [5, 6]. The automatic parking system includes a gyro, an incremental wheel pulse transducer (WPT), and ultrasonic sensors to measure the heading angle of the vehicle, traveling distance, and rear obstacle, respectively. The parking path is to navigate the vehicle to parking space when the vehicle reaches parking location. The parking location is calculated in advance according to
the corresponding parking space. The back-in parking is adopted in this study. The parking location of each parking space is determined according to the values of $n$ and $m$ of back-in parking, which is shown in Fig. 6. The $n$ and $m$ are derived by eq. (1) and (2), respectively.

$$n = R_{\text{min}} - c - b_1$$  \hspace{1cm} (1)

$$m \geq R_{\text{min}} - \sqrt{R_{\text{min}}^2 - (R_{\text{min}} - b_1)^2} + b_0$$  \hspace{1cm} (2)

where the $R_{\text{min}}$, $c$, $b_1$, and $b_0$ are shown in Fig. 6.

![Diagram of PT scheme and flow chart](image)

(a) An example of PT scheme (b) The flow chart of PT scheme

Figure 7. The proposed PT scheme

Path Tracking Scheme

The goal of the proposed path tracking (PT) scheme is to navigate the vehicle to track the driving path. The proposed PT scheme includes two phases, (1) determine goal point and (2) determine the steering angle, shown in Fig. 7, where the pink block shows the phase 1 and the phase 2 is shown in the blue block.

Initially, the RTK-GPS of vehicle is inputted to the PT scheme. Then, the phase is triggered to determine goal point. In the phase 1, all the RTK-GPS positions which are within the circle of radius $\Gamma$ (i.e., $\text{Dis} \leq \Gamma$) and located at the front of vehicle (i.e., $\Delta y > 0$) on the driving path are picked and stored to find the goal point. Once the goal point is determined, the phase 2 is triggered. In the phase 2, the goal point is inputted to pure pursuit algorithm [10] to calculate the curvature for the tracking from the current RTK-GPS position to the goal point. After the curvature is computed, curvature is substituted into the kinematic steering condition [11] to obtain the steering angle. The steering angle then is set as the input of Fuzzy-PID [12] to control steering wheel.

The RTK-GPS position is composed of $(x, y, h)$, where $x$ means the latitude, $y$ is the longitude, and $h$ is the heading of the vehicle. The pure pursuit algorithm uses the relations of the
relative horizontal distance \((H)\) and the distance \((D)\) between current RTK-GPS position and goal point to compute curvature \(\rho\).

\[
\rho = \frac{2H}{D^2}
\]  
(3)

Since the experimental vehicle uses the idle speed to park, i.e., the slip angles are zero, the steering angle can be determined by the kinematic steering condition. If we assume that a front-wheel-steering 4WS vehicle is turning to left on a circle of radius \(R\), the kinematic steering condition is expressed by,

\[
R = \sqrt{a^2 + l^2 \cot^2 \delta}
\]  
(4)

where

\[
\cot \delta = \frac{\cot \delta_1 + \cot \delta_2}{2}
\]  
(5)

The \(\delta_1\) and \(\delta_2\) represent the steer angle of the inner and outer wheels, respectively. The \(l\) is the distance between the front and rear axles. The \(a\) is the distance between the mass center of vehicle and rear axle. The \(\delta\) is the steering angle.

Then, the steering angle can be obtained by rewriting the eq. (4).

\[
\delta = \cot^{-1}\left(\frac{(R^2 - a^2)^{1/2}}{l}\right)
\]  
(6)

where

\[
R = \frac{1}{\rho}
\]  
(7)

The \(\rho\) represents the curvature of the circle with radius \(R\).

Therefore, in the PT scheme, when the curvature \(\rho\) is calculated by pure pursuit algorithm, the curvature \(\rho\) is substituted into the eq. (6) to obtain the steering angle.

**Implementation Results**

Our system is built on an experimental vehicle, Mitsubishi Savrin, which is shown in Fig. 8. The experimental parking lot is at a rectangular area with 35.5 x 31.5 m\(^2\), where there are 16 parking spaces (please refer to Fig. 4(a), where we assume that the first five parking spaces from the upper right are unoccupied). The size of each parking space is 6.04 x 2.4 m\(^2\).
The comprehensive implementation scenario is shown in Fig. 9 step by step. Initially, the RTK-GPS on the vehicle is turned on to obtain the position of the vehicle for the use of PT scheme. A PC is set up on the vehicle to wait a message from tablet. All the cameras also have been turned on before the implementation. Server builds the parking lot map according to the information from cameras.

Driver drives the vehicle to the entrance of the parking lot (see Fig. 9(a)). Driver takes his tablet and gets out the vehicle (see Fig. 9(b)). Driver then requests server to acquire the parking lot map. Driver touches the tablet to pick a parking space (see Fig. 9(c)). Once a parking space is selected (the first parking space from the right is picked in this experiment), tablet delivers a message to notify server that the parking space is occupied. After server receives this message, it updates parking lot map (see Fig. 9(d)). At the same time, tablet delivers a message to the PC on the vehicle for triggering the operation of parking. When the PC on the vehicle receives the message from tablet, the PC commands the MCU of vehicle to perform the proposed PT scheme (see Fig. 9(e). The PT scheme guides the vehicle to track the driving path until vehicle reaches the parking location.

As vehicle reaches the parking location of the selected parking space, it performs the back-in parking, where the steering wheel turns to maximum angle (see Fig. 9(f-g)). If the heading of vehicle is orthogonal with the heading of vehicle at the parking location, vehicle turns back the angle of the steering wheel to zero (see Fig. 9(h)). Then, the vehicle is backed straight until the rear ultrasonic sensor of the vehicle detects the warning triangle (see Fig. 9(i-j)). The stop threshold between vehicle and warning triangle is set as 1m. Figure 9(i-j) show the vehicle can be parked in the parking space successfully. When driver touches the tablet to retrieve his car, the parking lot map is updated and the vehicle is triggered to leave the parking space simultaneously (see Fig. 9(k-l)). The vehicle then automatically drives to a specific zone along with the driving path for the retrieving of user.

To measure the robustness of the proposed APS, we perform APS 20 times to observe the values of heading and \( n \) at the parking location. The experimental results indicate that the heading error can be controlled within \( \pm 1^\circ \) and the error of the \( n \) can be controlled within 15cm. Moreover, the vehicle can be parked into parking space successfully in each time.
(a) The vehicle is stopped at the entrance of the parking lot

(b) Driver gets out the vehicle and requests server to acquire parking lot map

(c) Driver selects a parking space

(d) Once the parking space is selected on the tablet, the server updates the parking lot map

(e) Vehicle starts to move from entrance to parking location

(f) Vehicle starts to perform back-in parking when it reaches the parking location

(g) Vehicle performs back-in parking

(h) Vehicle turns the steering wheel back

(i) Vehicle enters the parking space

(j) Vehicle finishes the back-in parking

(k) Driver selects parking space to retrieve his car

(l) Vehicle leaves the parking space

Figure 9. Implementation of autonomous parking system in the parking lot
Conclusions and future work

In this work, an autonomous parking system (APS) is implemented in a parking lot. The APS composed of cameras, server, tablet, and vehicle. The vehicle is equipped with two PCs, MCU, ultrasonic sensor, RTK-GPS, wheel speed sensor, and EPS. A path tracking (PT) scheme is proposed to guide the vehicle to reach the parking location of the selected parking space. Besides, the image recognition technique is realized to detect parking spaces. A complete wireless communication system is also built between camera, server, tablet, and vehicle. Implementation results indicate that our proposed APS can be realized in a parking lot successfully.

In the future, we add the rear camera on the vehicle to assist back-in parking. The images from the camera are used to calculate the distance between vehicle and the rear line of the parking space. Moreover, the obstacle detection module also will be added into the proposed autonomous parking system.

Acknowledgments

This work was supported by Ministry of Economic Affairs of Taiwan, under Grant Number 103-EC-17-A-25-0843.

Reference


