

特集 | Real time Mapping, Path Planning and Control for Autonomous Parking*

Hossein Tehrani

米 陀 佳 祐

Quoc Huy Do

三 田 誠 一

Keisuke YONEDA

Seiichi MITA

We present an architecture for autonomous parking at complex environment with narrow passages. A modified FastSLAM algorithm is developed to map environment by laser scanner. The proposed mapping method reduces the map entropy and increases the mapping and localization accuracy for safe autonomous parking. We also propose a real time path planning method to avoid static and moving obstacles and generate smooth movement for vehicle. The corresponding control commands for steering and acceleration are generated to minimize the heading and positioning errors during parking. The method is implemented and tested on autonomous vehicle platform for complicated parking in narrow passages. The experiments prove that the proposed method is able to do smooth and safe autonomous parking in real time at dynamic and complex parking environment.

Key words : Path Planning, Automatic Parking, Vehicle Control, SLAM, Mapping, Autonomous Driving

I. INTRODUCTION

Several automotive manufacturers (e.g., Toyota, Audi, BMW) have initiated the concept of “self-parking car” whereby a driver positions the automobile into a location that can be maneuvered into a parking spot under automated control by computer. On account of wide demand for autonomous parking, many parking control algorithms have already been developed¹⁾²⁾³⁾. An autonomous parking assistant generally performs complicated tasks including environment mapping, path planning and path tracking. The path tracking is combined of two components: a local path planner that generates a short range (a local path) to follow a globally Planned Path and a low level Motion (e.g. accelerating, steering, gear...) controller.

Autonomous vehicle requires a robust, reliable and accurate path tracking (control) method, that minimize the error of path tracking and satisfy the vehicle’s dynamics constraints. After parking path generation by parking management system, a set of motion control commands must be adjusted in a continuous manner to track parking path precisely. Control adjustment actions are formed by incorporating information from onboard sensing devices such as GPS, Inertial Magnetic Unit (IMU), Odometer and Laser Sensors. We extract current status of vehicle and parking path information such as position, heading angle and velocities from on board sensors. These data provide inputs to the path following controller

along the parking path that should to be followed. The controller processes these data to generate appropriate control signals to the actuators to follow the parking path and compensate errors. This process is executed repeatedly in order to provide closed-loop path tracking.

To perform path tracking, many algorithms have been proposed in literature. In¹⁾, Pradalier proposed a parking assistance system with help of a database for storing the pre-calculated control profile parameters and the resulting movement. However, it is heavy for an on-board system to maintain such a big database and control resolution is limited considering the computing resource constraints. In²⁾, Paromtchik presented an iterative algorithm for parking maneuver based on ultrasonic range data processing and sinusoidal function to control the steering angle and longitudinal velocity. However this method is only applicable for simple parallel parking maneuver. Kondak³⁾ applied numerical methods and artificial potential field to solve the nonlinear optimization problem of autonomous parking though it is time consuming and not suitable for real time. In⁴⁾, Wada presented a multilevel driver assistance system based on path planning and human interface to assist parking. Muller⁵⁾ proposed a continuous curvature trajectory design and feed-forward control for parking method based on two steps path planning. Lee⁶⁾ proposed a method to combine an open loop path planning and feed-

* © [2013] IEEE. Reprinted, with permission, from IEEE provided by Copyright Clearance Center with License number 3190630188184, July 16, 2013

This paper was published in IEEE Conference on Intelligent Transport System (ITSC), USA, 2012
Current status of vehicle and parking path information

back tracking controller for autonomous parking. However, the low level motion controller was ignored in his research.

In ⁷⁾, a four steps command planning method was proposed to park a CyCab car. However the results are depended on the accuracy of the servo-systems that execute the planned commands. Several algorithms utilized fuzzy logic method for vehicle parking control ^{8) 9)}. Though these methods are not easy to implement in practice because of nonlinear and computationally expensive conversion equations. Many researches concentrated on tracking and posture stabilization ^{10) 11)}. But results came back to be computationally expensive and takes a long time to stabilize car.

To overcome above mentioned problems including expensive computational costs, we proposed a fast and robust method for real time path generation and optimal steering control. The local path is a circular arc that connects the vehicle current position and heading to next goal point in map by considering the vehicle kinematic model and constraints. Arc shape paths generated very fast and can be followed precisely to solve the problem of expensive computation cost of tracking the parking path. We are able to react and pass moving objects such as pedestrians and cars in parking environment. We also proposed a command controller for steering angle to provide the minimum control error rate. By taking input from local path planner, the motion controller generates signal to control the vehicle accurately along the parking path.

The accuracy of environment mapping is a key factor for safe path planning and vehicle control in autonomous parking. FastSLAM (Fast Simultaneous Localization and Mapping) is already proposed as a novel method for real time mapping and localization ^{12) 13)}. However, enabling real-time SLAM in an unstructured large-scale parking environment is still a great challenge ¹⁴⁾. In this paper, we present a method to reduce the map entropy and generate accurate map for path planning and precise localization in narrow passages and complicated parking area. The proposed method is implemented on autonomous vehicle and we did some experiments at indoor and outdoor parking area to evaluate the efficiency and robustness of the proposed method.

Paper structure is as follow: Section II explains the autonomous parking architecture. Section III describes map-

ping parking area. Section IV presents the local path planning method. Section V presents the vehicle control model. Experimental results are presented in Section VI. Finally we conclude the research in Section VII.

II. System Overview

The architecture and flow chart of proposed autonomous parking is shown in Fig. 1. The autonomous parking has been done mainly in two steps as explained in following.

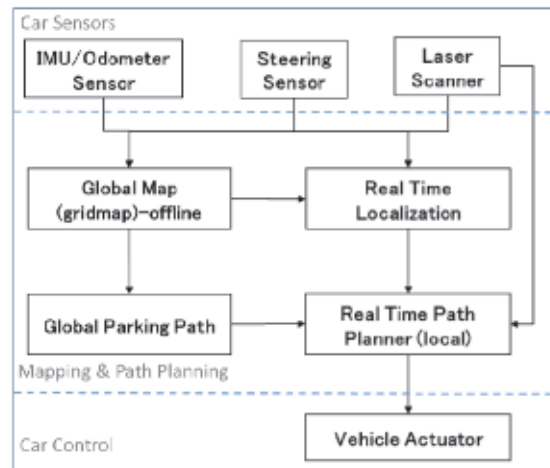


Fig. 1 Autonomous parking architecture and flowchart

Step1. Global Map Generation

We need a global reference map for path planning and vehicle control. We use occupancy grid map and improved FastSLAM algorithm to generate global map. The vehicle sensors data including laser data and IMU/odometer have been used to generate the global map as an occupancy grid map. The global map is prepared offline in advance by driving the vehicle in parking area. We record the laser and IMU/odometer data from the vehicle sensors and generate the global map as explained in Section III.

Step2. Real Time Path Planning and Vehicle Control

We use the generated global map as a reference for localization, path planning and control in autonomous parking. As shown in Fig. 1, we use the vehicle onboard sensors for real time localization in the global map. We assume the global path and parking goal points are given by higher level of parking management system. The localization and global parking path information have been used for real time local path generation from start to the parking area and deal

with the moving obstacles. Based on the given global parking path, we generate local paths for vehicle control at each timestamp of the vehicle control (100 milliseconds). Once the intended local path of vehicle has been generated by local path planner, the appropriate low level commands have been generated for vehicle actuators.

In this paper, we mainly focus on global map generation, local path planning and motion control for tracking the global parking path. The detail of global parking path planner and localization algorithm have been already explained in ^{15) 16)}.

III. Mapping Parking Area

A. Occupancy Gridmap Generation

Laser-based localization problem is similar to recognition problem as the autonomous vehicle seeks to match the current local map with a global reference map. Similar to SLAM algorithm ¹²⁾, the global reference map for parking is presented using occupancy grid and laser data are projected into two-dimensional square cells. The vehicle state X_t at time t is expressed by the following vector;

$$X_t = (x_t, y_t, z_t, \psi_t, \varphi_t, \theta_t, \delta_t) \quad (1)$$

- (x_t, y_t, z_t) : vehicle position at global map at time t ,
- $(\psi_t, \varphi_t, \theta_t)$: yaw, roll and pitch of vehicle at time t ,
- δ_t : front tire angle at time t ,

In probabilistic form, SLAM problem requires that the probability distribution $P(X_{0:t}, m | Z_{0:t}, U_{0:t}, X_0)$ be computed. This probability distribution describes the joint posterior density of the map m and vehicle state X_t (at time t) given the recorded observations $Z_{0:t}$ and control inputs $U_{0:t}$ up to and including time t together with the initial state X_0 of the vehicle. For simplicity, the joint SLAM state may be factored into a vehicle component and a conditional map component as shown in equation(2).

$$P(X_{0:t}, m | Z_{0:t}, U_{0:t}, X_0) = P(m | X_{0:t}, Z_{0:t}) P(X_{0:t} | Z_{0:t}, U_{0:t}, X_0) \quad (2)$$

This is a key property of FastSLAM algorithm, and the reason for its speed. The map is represented as a set of independent Gaussians, with linear complexity, rather than a joint map covariance with quadratic complexity. The essential structure of FastSLAM, then, is a Rao-Blackwellised (R-B) state ¹³⁾, where the trajectory is represented by weighted samples and the map is computed analytically.

The joint distribution, at time t , is represented by the set

$\{\omega_t^{(i)}, X_{0:t}^{(i)}, P(m | X_{0:t}^{(i)}, Z_{0:t})\}_t^N$ particles. At each time-step Δt , particles are drawn from a proposal distribution which approximates the true distribution, $P(X_t | X_{0:t-1}, Z_{0:t}, u_t)$.

The driving force is measured from odometer/IMU and steering potentiometer sensors from t until $t + \Delta t$ in the form $U_t = (x_{odo}, u_\psi, u_\varphi, u_\theta, u_\delta)$, where;

- x_{odo} : traveled distance from t until $t + \Delta t$ by odometer,
- $(u_\psi, u_\varphi, u_\theta)$: yaw, roll and pitch angular velocity from IMU.
- u_δ : tire angular velocity from steering potentiometer.

The actual angular velocities differ from the measured from the IMU and steering potentiometer due to noise. We model these differences by a zero-centered random variable with finite variance. If the vehicle state X_{t-1} is in the form $X_{t-1} = (x_{t-1}, y_{t-1}, z_{t-1}, \psi_{t-1}, \varphi_{t-1}, \theta_{t-1}, \delta_{t-1})$ sampling algorithm for $X_t \approx \pi(X_t | u_t, X_{t-1})$ is presented in (3);

$$\begin{aligned} \hat{\delta}_t &= \delta_{t-1} + \Delta t \cdot \hat{v}_\delta & \hat{v}_\delta &= u_\delta + \beta_1 \cdot \text{sample}(\sigma_\delta) \\ \hat{\psi}_t &= \psi_{t-1} + \Delta t \cdot \hat{v}_\psi & \hat{v}_\psi &= u_\psi + \beta_2 \cdot \text{sample}(\sigma_\psi), \\ \\ \hat{\varphi}_t &= \varphi_{t-1} + \Delta t \cdot \hat{v}_\varphi & \hat{v}_\varphi &= u_\varphi + \beta_3 \cdot \text{sample}(\sigma_\varphi), \\ \hat{\theta}_t &= \theta_{t-1} + \Delta t \cdot \hat{v}_\theta & \hat{v}_\theta &= u_\theta + \beta_4 \cdot \text{sample}(\sigma_\theta), \\ \hat{x}_{odo} &= x_{odo} + \beta_5 \cdot \text{sample}(\sigma_{odo}) \end{aligned}$$

$$\begin{bmatrix} \hat{x}_t \\ \hat{y}_t \\ \hat{z}_t \end{bmatrix} = \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ z_{t-1} \end{bmatrix} + [C_b^R(\hat{\psi}_t, \hat{\varphi}_t, \hat{\theta}_t)] \begin{bmatrix} \hat{x}_{odo} \cos(\hat{\delta}_t) \\ \hat{x}_{odo} \sin(\hat{\delta}_t) \\ 0 \end{bmatrix} \quad (3)$$

Return $(\hat{x}_t, \hat{y}_t, \hat{z}_t, \hat{\psi}_t, \hat{\varphi}_t, \hat{\theta}_t, \hat{\delta}_t)$

where $\text{sample}(\sigma)$ is normal or triangular distribution with zero mean and variance σ . $[C_b^R]$ shows the transformation matrixes for transforming from local to global coordination ¹⁷⁾.

The R-B particle filter for making map is as follows. We assume that, at time $t-1$ the joint state is represented by

$$\{\omega_{t-1}^{(i)}, X_{0:t-1}^{(i)}, P(m | X_{0:t-1}^{(i)}, Z_{0:t-1})\}_t^N \quad (4)$$

- For each particle draw a sample using (3);

$$X_t^{(i)} \sim \pi(X_t | u_t, X_{t-1}^{(i)}) \quad (5)$$

- Calculate the importance weight of samples according to the importance function as following;

$$\omega_t^{(i)} = \omega_{t-1}^{(i)} \cdot P(z_t | X_{0:t}^{(i)}, Z_{0:t-1}) \quad (6)$$

$$P(z_t|X_{0:t}, Z_{0:t-1}) = \int P(z_t|x_t, m) \cdot P(m|X_{0:t-1}, Z_{0:t-1}) dm \quad (7)$$

- If necessary, perform resampling.

B. Map Enhancement

The maps generated by general R-B are not accurate enough for autonomous parking in narrow spaces such as inside buildings and parking area of supermarkets. We propose a method based on the covariance of particles to update the global map $P(m|X_{0:t}, Z_{0:t})$. We update the laser observations in global maps by considering the vehicle position uncertainty as follows;

$$P(m|Z_{0:t}, X_{0:t-1}) = \int P(m|x_t, z_t) \cdot P(x_t|Z_{0:t}, X_{0:t-1}) dx_t \quad (8)$$

To calculate and update the global map based on the all vehicle states presented by the $X_t^{(i)} = (x_t^{(i)}, y_t^{(i)})$, $i = 1, \dots, N$, particles is time consuming and expensive. In this paper, we estimate the vehicle state distribution based the weighted covariance of the vehicle states as the following;

$$\bar{x}_t = \sum_i \omega_t^{(i)} \cdot x_t^{(i)}, \bar{y}_t = \sum_i \omega_t^{(i)} \cdot y_t^{(i)} \quad (9)$$

$$\sigma_t^x = \sum_i \frac{(x_t^{(i)} - \bar{x}_t)^2}{n-1}, \sigma_t^y = \sum_i \frac{(y_t^{(i)} - \bar{y}_t)^2}{n-1} \quad (10)$$

We update the global map considering nine states for vehicle as shown in **Table 1**. We compare the proposed method with general FastSLAM by evaluating the map entropy. The entropy of a single cell in the occupancy grid is the entropy of a discrete random variable with two possible outcomes, i.e. a Bernoulli distribution. Notice that the maximum entropy is obtained for $p(x,y)=0.5$, that is, for unobserved cells. Recalling the independency assumption between cells, the entropy for the whole map turns into (11):

$$H(\text{map}) = \sum_{x,y} -p(x,y) \cdot \ln(p(x,y)) - (1-p(x,y)) (1-\ln(p(x,y))) \quad (11)$$

$$\bar{H}(\text{map}) = \begin{cases} H(\text{map})/N_e & \text{if } N_e > 0 \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

where N_e represents the number of observed cells in the map, i.e. cells with an occupancy likelihood different from 0.5. We have done some experiments at indoor and outdoor parking area to evaluate the map entropy. **Fig. 2** shows global mapping of the outdoor parking environment using autonomous vehicle equipped with SICK laser scanner. As

shown in **Fig.2 (a), (b)**, the parking map has been improved using new approach for map updating. The results of the experiments for are shown in **Table 2**. As shown in **Table 2**, the entropy of the generated maps are considerably improved compared to general FastSLAM.

Table 1. Vehicle nine status for updating map

$(\bar{x}_t - \sigma_t^x, \bar{y}_t - \sigma_t^y)$	$(\bar{x}_t, \bar{y}_t - \sigma_t^y)$	$(\bar{x}_t + \sigma_t^x, \bar{y}_t - \sigma_t^y)$
$(\bar{x}_t - \sigma_t^x, \bar{y}_k)$	(\bar{x}_t, \bar{y}_k)	$(\bar{x}_t + \sigma_t^x, \bar{y}_k)$
$(\bar{x}_t - \sigma_t^x, \bar{y}_t + \sigma_t^y)$	$(\bar{x}_t, \bar{y}_t + \sigma_t^y)$	$(\bar{x}_t + \sigma_t^x, \bar{y}_t + \sigma_t^y)$

Table 2. Gridmap Entropy Comparison

No	Environment	Mapping Area (m × m)	Grid map Entropy	
			FastSLAM	Proposed Approach
1	Indoor parking	42.25×30.10	0.0260529	0.00142978
2	Indoor parking	27.15×24.25	0.0138635	0.000807971
3	Outdoor parking	100.05×84.50	0.0275403	0.00792302
4	Outdoor parking	161.30×91.00	0.0174873	0.007309377

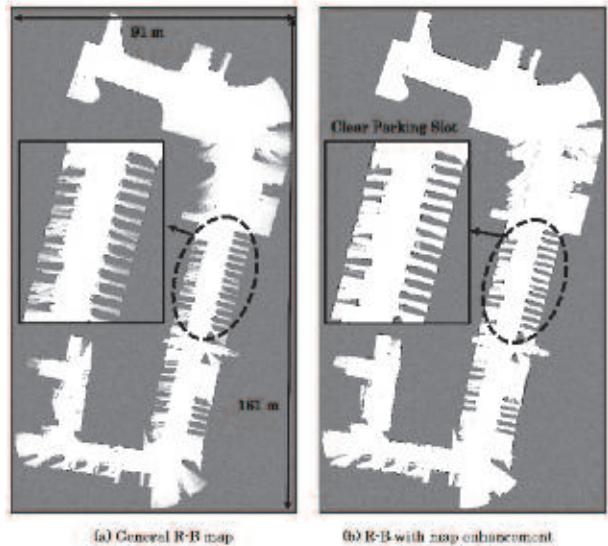


Fig. 2 Global mapping the outdoor parking space by autonomous vehicle

IV. Path Planning

A. Vehicle dynamic model

The following simple kinematic model is considered to generate local paths,

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{\delta} \end{bmatrix} = \begin{bmatrix} \cos \psi \\ \sin \psi \\ \frac{\tan \delta}{l} \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} w \quad (13)$$

where (x,y) are 2D coordinate of vehicle rear axis's middle point, ψ measures the heading angle (the orientation of the vehicle respect to the X axis), δ is the steering angle, v and w are the longitudinal and rotational velocity of the vehicle, l is the distance between the front and rear wheel axis. The original vehicle kinematic model shown in (13) works in a global coordinate system, which is difficult to keep high accuracy according to the accumulation errors and noise in sensors observations and measuring. It is more precise to change to local coordinate system for smooth and stable vehicle control.

Fig. 3. shows the definition of a local coordinate system for vehicle control. Let d be the distance between the vehicle rear axle midpoint and the desired path, naming it as P , s is the corresponding arc length that the point P moves along the path, inclination angle of the tangent line at point P as ψ_t , $\psi_t = \psi - \psi_p$.

The curvature $c(s)$ along the path is defined as following:

$$c(s) = d\psi_t/ds \quad (14)$$

which can be rewritten as $d\psi_t = c(s) \cdot ds$. Assume that $c(\cdot) \in C^1$, and the path satisfies the following conditions;

$$\begin{aligned} \dot{s} &= v \cos \psi_p + \dot{\psi}_t d \\ \dot{d} &= v \sin \psi_p \end{aligned} \quad (15)$$

By substitute (15), (14) into (13), the local kinematic equation for vehicle control derived as (16) with the local path coordinate $q_p = (s,d,\psi_p,\delta)$ that can be acquired by on board sensors of vehicle platform without any observer design.

$$\begin{bmatrix} \dot{s} \\ \dot{d} \\ \dot{\psi}_p \\ \dot{\delta} \end{bmatrix} = \begin{bmatrix} \frac{\cos \psi_p}{1-d \cdot c(s)} \\ \sin \psi_p \\ \frac{\tan \delta}{l} - \frac{c(s) \cos \psi_p}{1-d \cdot c(s)} \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} w \quad (16)$$

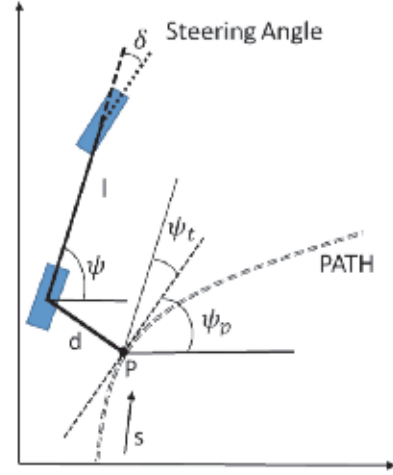


Fig. 3 Coordination system for path following task

B. Local Path Planner

The local path planner is a reactive process to avoid dynamic and static obstacles and relies on the latest sensor data to maintain vehicle safety and stability. By a given global parking path, we call the local path planner frequently after a short timestamp to generate paths for vehicle movement. The local path planner should have high performance and quick response in dynamic environment. We generate a set of pre-defined smooth paths with considering vehicle turning constraints as the following:

$$P = \{\rho_i | \rho_i = f(\delta_i) | i = 1, \dots, C\} \quad (17)$$

where δ is the steering angle which is limited by the vehicle designed maximum steering angle: $\delta_{min} \leq \delta \leq \delta_{max}$; C is defined by the maximum steering value and the steering angle resolution. Fig. 4 illustrates the local environment with a set of predefined paths, local goal point and the selected local path to follow the global path and avoid obstacle. Function $f(\delta_i)$ maps each steering angle δ_i to a specific set of pre-calculated path points in (18),

$$\rho_i = \{(x_i^j, y_i^j) | j = 1 \dots, N\} \quad (18)$$

where N is the path points number. These path points are generated based on the vehicle kinematic model in (16). Let $D(\rho)$ be the closest distance between the local goal point and the pre-defined path ρ and P_{free} is the set of obstacle-free paths. The final path is selected to minimize the $D(\rho)$ and gradient of steering angle (smooth steering) from P_{free} set, as following;

$$\rho_{final} = \min_{\delta, D(\rho)} \rho(\delta), \forall \rho(\delta) \in \rho \in P_{free} \quad (19)$$

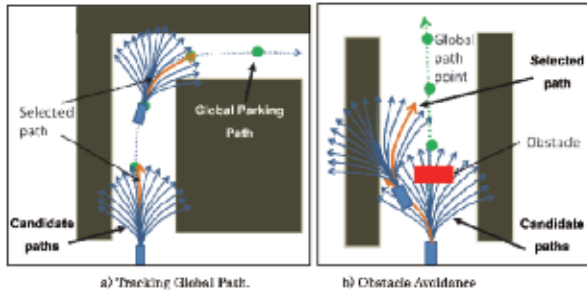


Fig. 4 Local path planner

Fig.5. depicts the local path planner during autonomous parking experiments. The local path is selected to follow the global parking path and minimize the curvature of the vehicle.

The proposed local path planning method is effective not only for tracking the global parking path but also for smooth movement in the narrow space through dynamic obstacles.

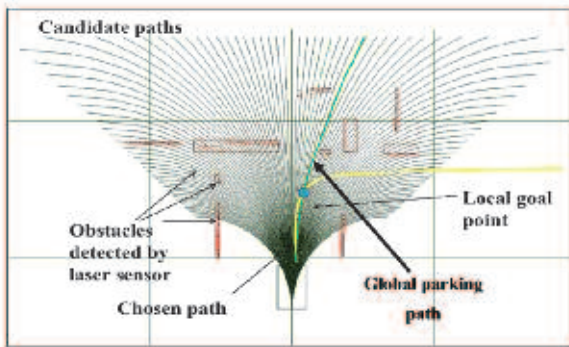


Fig. 5 Local path planner screen during autonomous parking

V. Vehicle Control Model

Generally, the vehicle actuators are divided into two main parts including the velocity actuator and the steering control. We have already proposed a method for optimal velocity control in ¹⁸⁾. Here we propose steering control method to minimize the drift of steering in narrow passages area.

In steering control module, a follow-up control system is applied to achieve any given steering angle. Due to the simplification, the steering controller is modeled in three working modes, i.e. dead zone, working zone and saturation mode. It is reasonable to ignore the time delays in the amplifier without losing too much accuracy. The voltage command to the steering motor is manipulated by the AD/DA converter from control computer. We use the hyperbolic tangent function as shown in Fig. 6 to estimate the AD/DA

output signal value according to the desire steering angle.

$$U(t) = \frac{1}{2} \left[(a + b) + (b - a) \tanh \left(\frac{\delta(t) - t_0}{w} \right) \right] \quad (20)$$

where a and b are system limited range signal values, t_0 is the "break" point (i.e., the place where the slope changes from increasing to decreasing) and w is the "width" or steering angle constant. We are trying to minimize the drift of the steering angle in (26)

$$\min d = \int_0^T (\delta(t) - \hat{\delta}(t))^2 dt \quad (21)$$

where $\delta(t)$ is desired steering angle and $\hat{\delta}(t)$ is the actual steering angle. The constraints consist of boundary of the variables, such as

$$\delta_{min} < \delta_t < \delta_{max}, U_{min} < U < U_{max} \quad (22)$$

We have estimated the parameters in (20) to minimize the (21) through the experiments.

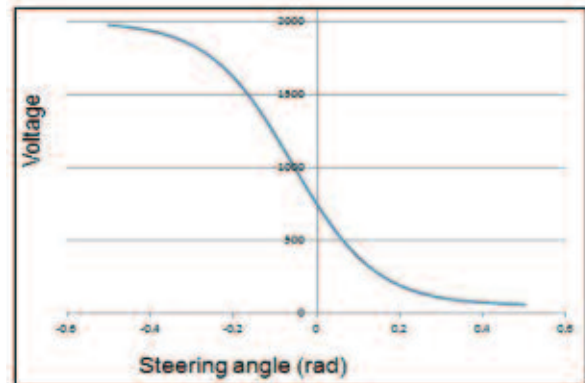


Fig. 6 AD/DA Output signal estimation by Hyperbolic tangent

VI. Experiment

We have done some experiments in indoor and outdoor parking area to evaluate the proposed method for autonomous parking. Generally in Japan, the parking area and slots are narrow and we have selected such narrow parking area for experiments. The parking goal point and global path are given in advance though it can be extracted automatically using camera or information from parking management system.

A. Autonomous Vehicle Platform

We use an electric car which equipped with an onboard array of sensors and computers as shown in Fig. 7. Our

vehicle is Small Electric Vehicle (SEV) which enhanced for fully autonomous driving. A DC motor is installed to control the steering wheel for autonomous driving. Control computer sends the voltage signal to the analog positional control and it generates suitable command to DC servo motor driver. Vehicle Electric Control Unit (ECU) controls the engine torque and position of accelerator throttle by information from accelerator sensor. The control computer controls the accelerator throttle by setting desire input voltage to the acceleration control of ECU. Finally, we utilize a compact ball screw linear motion actuator for control of vehicle brake throttle.

We have installed three potentiometers to read position and direction of the accelerator, brake throttle and steering wheel. Moreover, a rotary encoder is installed in a left rear wheel of the vehicle for measure the traveled distance and speed of the vehicle. Two laser scanners have been installed on vehicle for obstacle avoidance, environment mapping and local navigation. An Inertial Magnetic Unit (IMU) with GPS provides data from 3 accelerometers, 3 gyroscopes, a 3D magnetic field sensor and GPS information.



Fig. 7 Autonomous vehicle platform

B. Global mapping and path planning

We use proposed global mapping algorithm to prepare a map of the parking area. The generated occupancy grid global map is presented in Fig. 8 (a) with 5 cm cell resolution. We have prepared global map in advance before starting the autonomous parking experiments. We use the global path planner in ¹⁵⁾ to generate path from start to goal point. The global parking path points are shown in solid line in Fig 8 (a).

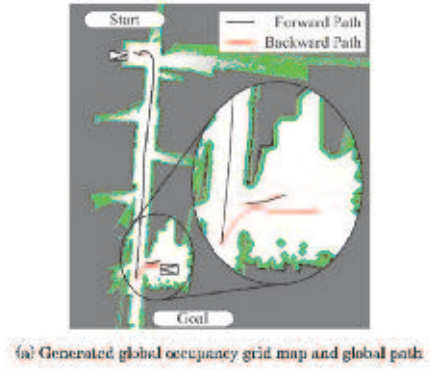


Fig. 8 Real time autonomous parking experiment

C. Steering Control

We have estimated and optimized the steering control parameters in equation (20) through the experiments. The following values are selected for parameters in equation (20) to generate control signals to AD/DA convertor and minimize the steering drift.

$$a = 2150, \quad b = -50, \quad t_0 = -0.0365, \quad w = 0.4$$

D. Autonomous Parking

We have already developed a precise localization algorithm based on multi observation particle filters ¹⁶⁾. We use the laser data to initialize the vehicle position in the global map. After localization, and generating global parking path, the local path planner generates local paths to the final parking point. The motion controller also generates steering and velocity commands to the vehicle actuators to track the parking path and minimize the error. The local path is updated every 100 milliseconds based on the vehicle current position and heading and it is able to handle and pass moving obstacle. The results of two experiments are shown in Fig. 9(a), (b). The solid line shows the designed global path for parking and the dashed line shows the vehicle run. As shown in Fig. 9 the proposed local path planner and motion control methods are able to accurately track the designed global parking path to the final parking point in narrow passages. The lateral distance error of designed global path and

actual vehicle run are shown in Fig. 10(a), (b). The average lateral distance error for experiments I and II are 0.1217 meter and 0.075 meter respectively.

The real time vehicle autonomous parking is shown in Fig 8 (b). The person in the vehicle is just for safety and all functions of vehicle control, path planning and localization are done autonomously based on the architecture presented in Fig. 1.

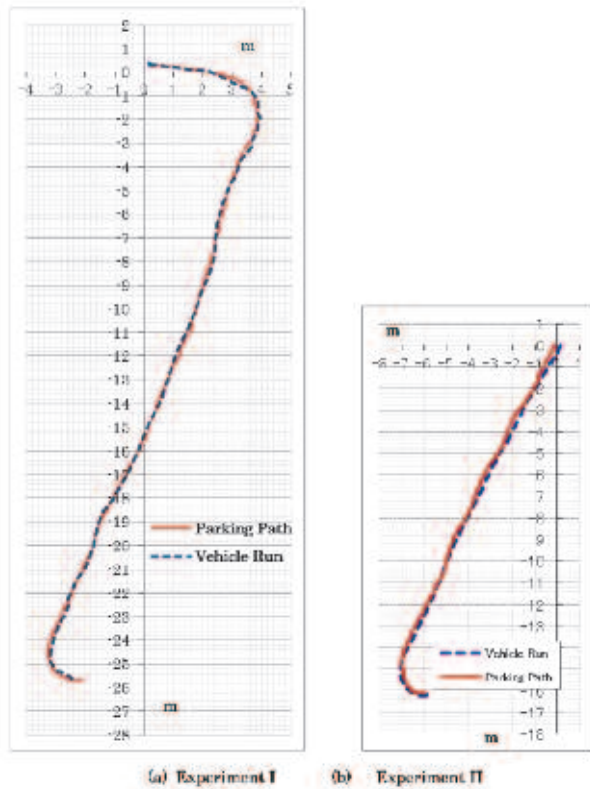


Fig. 9 Vehicle localization, path planning and control

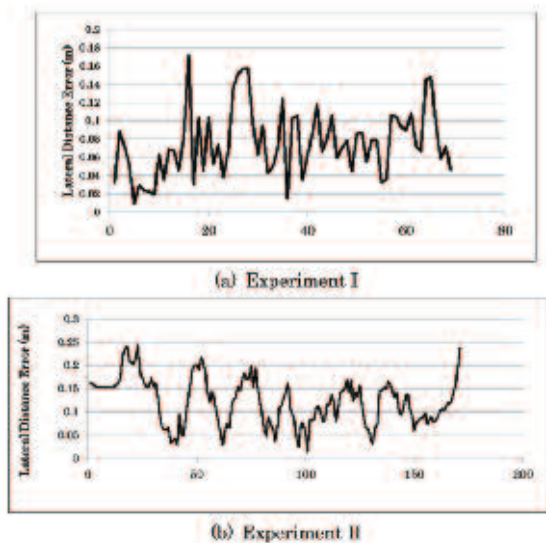


Fig. 10 Lateral distance error of designed global path and actual vehicle run

VII. Conclusion

We have presented a robust and practical architecture for autonomous parking. We have extended our previous work to consider parking area mapping and local path planning. The proposed method is real time and designed to handle moving obstacles and narrow parking passageway. We have implemented the parking algorithm on autonomous vehicle platform. Our experiments in narrow passage parking area proved the effectiveness of method for autonomous parking in complex and dynamic environment.

REFERENCE

- 1) C.Pradalier, S.Vaussier, and P.Corke, "Path planning for a parking assistance system: Implementation and experimentation." Sydney, Australia: Australian Robotics and Automation Association Inc, 2005.
- 2) I.E.Paromtchik and C.Laugier, "Motion generation and control for parking an autonomous vehicle," in *Proc. Of the IEEE Int. Conf.on Robotics and Automation*, pp. 3117–3122, 1996.
- 3) K. Kondak and G. Hommel, "Computation of time optimal movements for autonomous parking of non-holonomic mobile platforms." International Conference on Robotics and Automation, 2001.
- 4) M. Wada, K. S. Yoon, and H. Hashimoto, "Development of advanced parking assistance system", *IEEE Transactions on Industrial Electronics*, 2003.
- 5) B.Muller, J. Deutscher, S.Grodde, "Continuous curvature trajectory design and feedforward control for parking a car", *IEEE Transactions on Control Systems Technology*, pp. 541 – 553, 2007.
- 6) K. T. C. Lee, C. Y. Tsai, and K. T. Song, "Fast parking control of mobile robots: a motion planning approach with experimental validation", *IEEE Transactions on Control System Technology*, VOL. 12, NO. 5, 2004.
- 7) I.E.Paromtchik, "Steering and velocity command for parking assistance", *Proceeding of the 10th IASTED International Conference Robotics and Applications*, 2004.
- 8) T.-H.S. Li, C.S.Jie, "Autonomous fuzzy parking control of a car-like mobile robot", *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, pp. 451 – 465, 2003.

- 9) I.Baturone, F.J.Moreno-Velo, S.S.Solano, A.Ollero, “Automatic design of fuzzy controllers for car-like autonomous robots”, *IEEE Transactions on Fuzzy Systems*, pp. 447 – 465, 2004.
- 10) D. Gorinevsky, A. Kapitanovsky, and A. Goldenberg, “Neural network architecture for trajectory generation and control of automated car parking,” *Transactions on Systems Technology.*, vol. 4, pp. 50–56, 1996.
- 11) H. An, T. Yoshino, D. Kashimoto, M. Okubo, Y. Sakai, and T. Hamamoto, “Improvement of convergence to goal for wheeled mobile robot using parking motion,” in *IEEE International Conference Intelligent Robots Systems*, 1999.
- 12) M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit. “FastSLAM: A factored solution to the simultaneous localization and mapping problem“. Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA, USA, July 2003; pp. 593-598.
- 13) H. Durrant-Whyte, T. Bailey, “Simultaneous localization and mapping (SLAM): Part I, the essential algorithms.” *Robotic and Automation Magazine*, Vol. 13, No. 2, pp. 99-110, 2006.
- 14) T. Bailey , H. Durrant-Whyte , “Simultaneous localization and mapping (SLAM): Part II” *Robotic and Automation Magazine*, Vol. 13, No. 3, pp. 108-117, 2006.
- 15) L.Han, Q.H.Do, S.Mita, “Unified path planner for parking an autonomous vehicle based on RRT”, *International Conference on Robotics and Automation*, 2011.
- 16) H. Tehrani Niknejad, S. Mita and L. Han, “Multi-Sensor Data Fusion for Autonomous Vehicle Navigation through Adaptive Particle Filter,” in *Proc. IEEE Intelligent Vehicle Symposium (IV).*, Jun 2010, pp. 752–759.
- 17) E. Bekir, *Introduction to Modern Navigation Systems*, World Scientific Publishing, pp. 241-260, 2007.
- 18) L.Han, Q.H.Do, C.Guo, S.Mita, “Optimal Motion Generation for autonomous vehicle in Maze-like Environment”, *IEEE Intelligence Transportation System Conference*, pp. 1299-1304, October 2011.

<著 者>



Hossein Tehrani
研究開発3部 (工学博士)
インテリジェント ビークル
(軌道計画法, ナビゲーション),
センサーフュージョン,
コンピュータビジョン,
パターン認識, 機械学習



米陀 佳祐
(よねだ けいすけ)
豊田工業大学 ポストドクトラル
研究員 博士 (情報科学)
自律自動車の走行経路計画, 自己
位置推定技術及び機械学習に
関する研究に従事



Quoc Huy Do
豊田工業大学 博士後期課程
自律自動車の走行経路計画に
関する研究に従事



三田 誠一
(みた せいいち)
豊田工業大学 特任教授
(工学博士)
スマートビークル研究センター長
信号処理, 画像処理