Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Applications of Highly Accurate Localization and Navigation to Mobile Robot

Guo-Shing Huang
National Chin-Yi University of Technology
NCUT
Taichung, Taiwan
hgs@ncut.edu.tw

Jie-Cong Ciou
National Chin-Yi University of Technology
NCUT
Taichung, Taiwan
ngyma@yahoo.com.tw

Hsiung-Cheng Lin
National Chin-Yi University of Technology
NCUT
Taichung, Taiwan
hclin@ncut.edu.tw

***Abstract*- The optimal route planning is to seek the most appropriate path for robot to arrive the desired destination smoothly in the shortest time. In this paper, the Tangent method is firstly used to find the shortest moving route. Secondly, the Kalman Filtering algorithm is employed to amend the route errors at the k^{th} time during the moving status. Simultaneously, the route for next $(k+1)^{th}$ time can be also estimated. Finally, the robot route is continuously adjusted using the fuzzy logic controller for the robot moving more smoothly and efficiently. Both simulation and experimental results confirm that the robot can reach the destination fast within no exceeding ±2cm localization error.**

Keywords: Route Planning, Kalman Filtering, Fuzzy Logic Control, Ultrasonic Sensor, Laser Navigation System

I. INTRODUCTION

The optimal route planning for mobile robots is still a crucial research field [1-8]. The key research point is that the robot is able to correct its current position and estimate the next route to arrive at the destination accurately and quickly without collision. In recent years, a number of publications related to the navigation and route planning can be found in the literature, for instance, the Monte Carlo [1], histogram statistic [2], fuzzy control [3][4][5], the genetic algorithm[8], etc. In the past, the Kalman Filtering algorithm was mostly used in finding the posture degree of GPS or applied in such as inertia navigation, and information integration [9][10]. This paper uses the Tangent method to determine the robot's shortest path, and the Kalman filtering algorithm is applied to estimate the location of the robot. The fuzzy logic controller is designed to amend the navigating error. Accordingly, the optimal navigation planning for the mobile robot can be achieved.

This paper is divided five sections. Section II describes the motion model of the mobile robot as well as the model of laser-based location measurement. The Tangent method combining with the Kalman Filtering algorithm and fuzzy control is presented in section III. Section IV provides both simulation and experimental results to verify the effectiveness of the proposed approach. The conclusions are given in section V.

II. DESCRIPTION OF BOBOT MOTION AND MEASUREMENT MODEL

A. Model of Robot Motion

The model of robot motion is depicted in Fig.1. The safe range between the robot and the obstacle is defined as the radius r_R. The coordinates of the robot's current location and its target destination are defined as (x_R, y_R) and (x_G, y_G), respectively. In this model, a number of ultrasonic sensors were installed to measure the object (robot, target and obstacle) locations within the 180-degree range area in the front of moving robot.

The robot's heading angle is defined as θ_R. The angle θ_{RG} and the distance D_G between the robot and the target can be calculated from the equations (1) and (2), as follows.

$$\theta_{RG} = \tan^{-1}\frac{y_G - y_R}{x_G - x_R} \tag{1}$$

$$D_G = \sqrt{(x_G - x_R)^2 + (y_G - y_R)^2} \tag{2}$$

The distance and included angle between the robot and obstacle are defined as D_O and θ_{RO}, respectively. They can be also obtained similarly using the equations (1) and (2). The speed of robot motion (S) and its moving angle (θ), are restricted as the range of $S \in (0, 1.0)$ m/s and $\theta \in (-30°, 30°)$. Note that θ is positive when robot turns right, and θ is negative when the robot turns left.

The robot motion equations are shown as follows.

$$\Delta x_R = \Delta l \cos\theta_R \tag{3}$$

$$\Delta y_R = \Delta l \sin\theta_R \tag{4}$$

$$\Delta\theta_R = \theta_R(k+1) - \theta_R(k) \tag{5}$$

Note that l is the robot moving distance. Δx_R, Δy_R, $\Delta\theta_R$ and Δl is the changing amount in the abscissa, ordinate, moving angle and distance, respectively. Consider every change in $\Delta\theta_R$, i.e., from the k^{th} time to $(k+1)^{th}$ time so that the whole path of robot moving can be formed as a curve route.

978-1-4244-2794-9/09/$25.00

B. Model of Location Measurement

The model of location measurement is shown in Fig. 2. The laser navigation system was used to settle the reflect reference location, including the distance and angle. Therefore, the robot coordinate (x_R, y_R) and heading angle (θ_R) can be obtained using Triangular method. Note that (x_n, y_n) is the n^{th} reflect coordinate, and α_n is the angle of the vector $(\vec{v_n})$ from the robot (x_R, y_R) to (x_n, y_n). ρ_n is the length of $\vec{v_n}$ and α_{Rn} is the angle between v_n and robot heading direction.

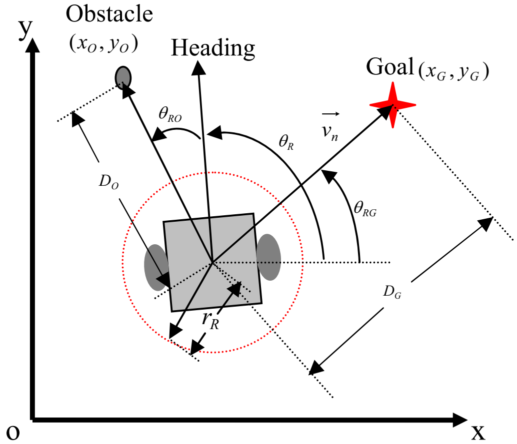

Fig.1 Model of Robot Motion

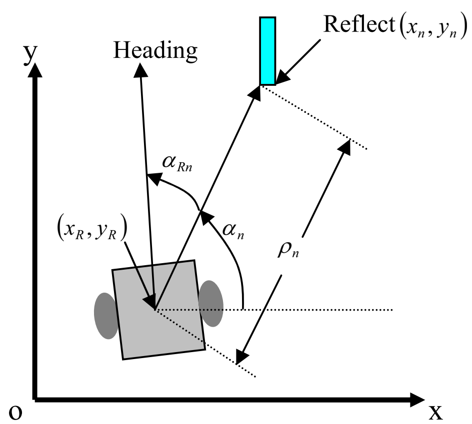

Fig.2 Model of Location Measurement

III. The route planning

To ensure the optimal path to be achieved for the moving robot, the Tangent method is firstly used to find the shortest route. The Kalman Filtering algorithm is then applied for the robot path estimation at next $(k+1)^{th}$ moment when the robot is moving ahead at the k^{th} moment. The fuzzy logic controller is to correct the moving route error and enables the $(k+1)^{th}$ moment achieving the minimum error. The flowchart of the proposed scheme for the route plan is as shown in Fig. 3.

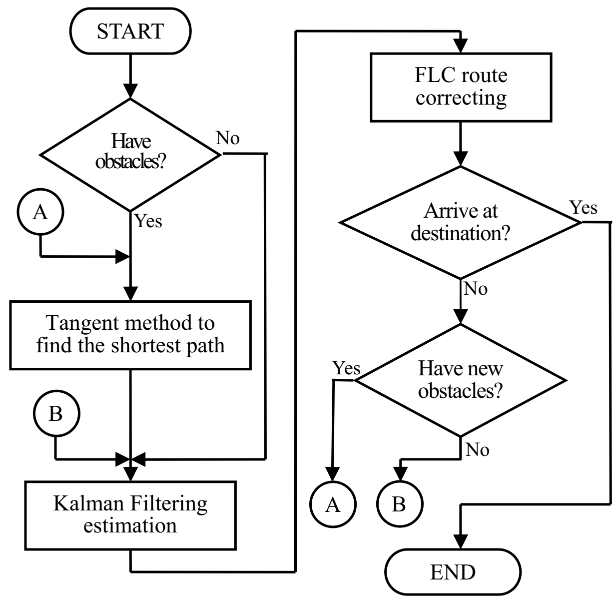

Fig.3 Flowchart of the route planning

A. The shortest route using the Tangent method

Consider the case with obstacles, the possible shortest route is shown in Fig. 4. The symbols R($=(x_R, y_R)$), B ($=(x_B, y_B)$) and G($=(x_G, y_G)$) are to represent the location coordinates for the mobile robot, the obstacle and the destination, respectively. The maximum radius range of the obstacle is defined as r_B. The straight line ($\overline{RG}$) between the robot and the destination can be described as follows.

$$\frac{y - y_R}{x - x_R} = \frac{y_R - y_G}{x_R - x_G} \tag{6}$$

The equation (6) can be rewritten as

$$(y_R - y_G)x + (x_G - x_R)y + (x_R y_G - x_G y_R) = 0 \tag{7}$$

The distance (d) from the obstacle (x_B, y_B) to $\overline{RG}$ can be obtained from the following equation.

$$d = \frac{\left|(y_R - y_G)x_B + (x_G - x_R)y_B + (x_R y_G - x_G y_R)\right|}{\sqrt{(y_R - y_G)^2 + (x_G - x_R)^2}} \tag{8}$$

If $d > r_B + r_R$, the robot can arrive at the destination along the straight line $\overline{RG}$ without collision. If $d \le r_B + r_R$, B is regarded as an unavoidable obstacle. Consequently, the radius $(r_B + r_R)$ can be formed as a collision circle, and its equation is obtained as follows.

$$(x - x_B)^2 + (y - y_B)^2 = (r_B + r_R)^2 \tag{9}$$

Therefore, the tangent line from the R (x_R, y_R) to the collision circle can be expressed as

$$(y - y_R) = m_1(x - x_R) \tag{10}$$

On the other hand, the tangent line from the collision circle to the G (x_G, y_G) can be expressed as

$$(y - y_G) = m_2(x - x_G) \tag{11}$$

The coordinates of c1 and c2 can be calculated from the equations (9) to (11) using the principles of geometry and trigonometry. Two possible shortest paths (R→c1→G and R→c2→G) can be thus obtained. Consequently, the shortest way can be determined, based on these paths.

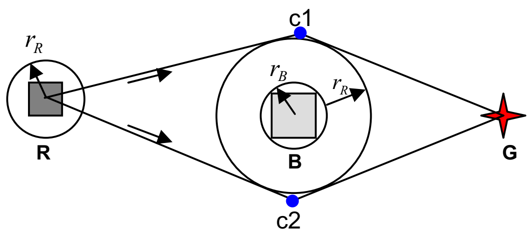

Fig.4 Route plan of the shortest Path

The above method can be extended to the case with more obstacles or new obstacles appearing when the robot is moving ahead. For example, when the robot finds another obstacle on the way, the coordinate of previous obstacle can be replaced by the new one and the new robot coordinate starts from this midway (location). The same procedure will be repeated until the robot arrives at the destination.

B. The Route Estimation using Kalman Filtering

Kalman Filtering is employed to estimate the next route at the $(k+1)^{th}$ time. The state variable is defined as

$$x = [\Delta v_{xR} \quad \Delta v_{yR} \quad \theta_R \quad x_R \quad y_R]^T \tag{12}$$

where is v_{xR} and v_{yR} are robot movement speeds at x axle and y axle direction, respectively. Define the measurement equation as

$$z = [\Delta x_R \quad \Delta y_R \quad \theta_R \quad x_1 \quad y_1 \quad \ldots \quad x_n \quad y_n]^T \tag{13}$$

Therefore, the observation equation can be written as follows.

$$z(k) = H(k)x(k) + e(k) \tag{14}$$

The system state equation is written as

$$x(k+1) = \phi(k)x(k) + \omega(k) \tag{15}$$

where k is the discrete time, $H(k)$ is measurement function matrix, $e(k)$ is the measurement error, and $\omega(k)$ is the white noise process. The measurement function matrix $H(k)$ and state transform matrix is described as follows.

$$H(k) = \begin{bmatrix} \Delta t & 0 & 0 & 0 & 0 \\ 0 & \Delta t & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{\rho_1 \cos\alpha_1}{x_R} + 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{\rho_1 \sin\alpha_1}{y_R} + 1 \\ & & & \vdots & \\ 0 & 0 & 0 & \frac{\rho_n \cos\alpha_n}{x_R} + 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{\rho_n \sin\alpha_n}{y_R} + 1 \end{bmatrix} \tag{16}$$

$$\phi(k) = \begin{bmatrix} 1 & 0 & 0 & \frac{\Delta x_R}{\Delta t \cdot \Delta v_{xR}} & 0 \\ 0 & 1 & 0 & 0 & \frac{\Delta y_R}{\Delta t \cdot \Delta v_{yR}} \\ 0 & 0 & \frac{\Delta \theta_R}{\theta_R} + 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{\Delta x_R}{x_R} + 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{\Delta y_R}{y_R} + 1 \end{bmatrix} \tag{17}$$

The robot moving location can be estimated according to the following steps.

1) Set the initial state: $\hat{x}(0 \mid 0) = x_0$ and initial variance-covariance matrix $P(0 \mid 0) = P_0$.

2) Estimate the state and variance-covariance matrix.

$$\hat{x}(k+1 \mid k) = \phi(k+1, k)\hat{x}(k \mid k) \tag{18}$$

$$P(k+1 \mid k) = P(k \mid k) + Q(k) \tag{19}$$

where $Q(k)$ is the process noise variance of $\omega(k)$.

3) Update and measure new state and state estimation error variance.

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + K(k+1)[z(k+1) - \hat{z}(k+1)] \tag{20}$$

$$P(k+1|k+1) = [I - K(k+1)H_x(k+1)]P(k+1|k) \tag{21}$$

where $K(k+1)$ is the Kalman Filtering gain.

$$K(k+1) = P(k+1|k)H_x^T(k+1)[H_x(k+1)P(k+1|k)H_x^T(k+1) + R(k+1)]^{-1} \tag{22}$$

where the $R(k+1)$ is the measurement noise covariance of $v(k+1)$. The state $\hat{x}(k+1|k+1)$ after filtering can be obtained so that the next robot route and moving angle can be estimated in advance.

C. Amendment of the moving path using Fuzzy Logic Controller

From Fig.4, the shortest path is assumed as R→c1→G. However, this path is found with an abrupt change at the c1 location. At this point in practice, the robot must stop a moment for the turning action before moving forward. This performance will normally take a long time and require a lot of power consumption. To resolve this problem, this study proposes the fuzzy logic control (FLC) to modify the robot moving path, enabling the robot moving smoothly and efficiently. The block diagram of the robot moving modification with FLC is shown in Fig.4. The θ_E is defined as the angle error between the robot current angle and desired angle. Two output variables, i.e., S_L S_R, from FLC are defined as the left wheel rotating speed and the right wheel rotating speed (S_R), respectively. The output of FLC is used to drive two DC motors using two PWM controllers.

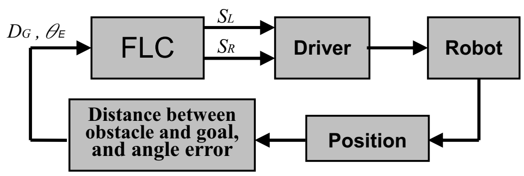

Fig.5 Block diagram of FLC scheme

The membership functions of the input and output variables θ_E, D_G, S_L and S_R are shown in Fig. 6, where the membership function of S_L and S_R is the same as D_G. The final output is determined using the look-up table which is stored in the database in advance. The fuzzy reasoning rule is shown in TABLE 1. Through the fuzzy reasoning, we can obtain a fuzzy quantity and select the center of gravity method to change fuzzy quantity into the actual output quantity.

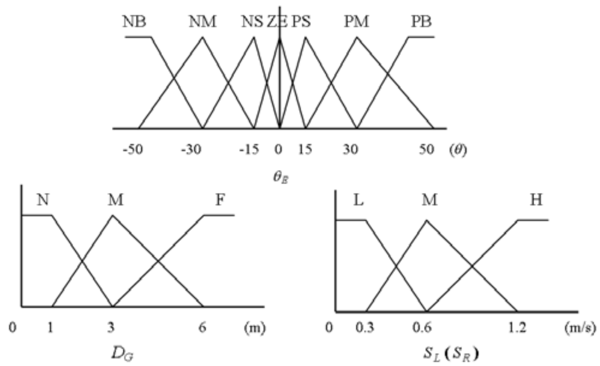

Fig.6 Membership function of Input and output

TABLE 1. Rule Table of FLC Fuzzy Reasoning

S_L / S_R		D_G N		D_G M		D_G F	
θ_E	**PB**	H	L	H	L	M	L
	PM	M	L	M	L	H	M
	PS	M	L	M	L	M	M
	ZE	L	L	M	M	H	H
	NS	L	M	L	M	M	M
	NM	L	M	L	M	M	H
	NB	L	H	L	H	L	M

IV.PERFORMANCE RESULTS

In order to verify the validity of the proposed algorithm, firstly we used the computer to run the simulation. Then, the practical implementation was carried out following up the same environment as the simulation.

A. Software Simulation Results

Initially, the Kalman Filtering combining the fuzzy logic controller to amend the route error was verified by MATLAB program. Note that the noise level was set up randomly within the measured signal of ±3%. Additionally, a sharp noise of localization error was added into the system at the 100-second and 200-second time domain. The simulation result is shown in Fig. 7. Clearly, it is found that the system can correct the error in 2 seconds. In the normal situation, the average error is ±1cm if adding no noises.

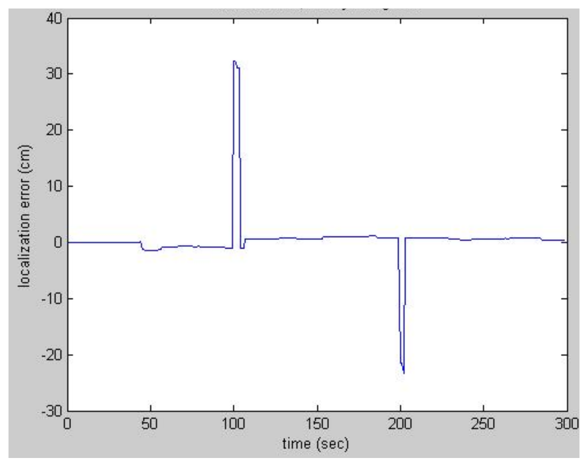

Fig.7 Localization error response

Secondly, the motion rule was simulated using Borland C^{++} Builder (BCB). Assume that the robot has self-locating and environmental cognition functions. It is to simulate laser navigation system, SICK NAV200, and laser scanners system, SICK LMS291. Also, we set up the moving speed as 0~0.5m/s, and the coordinate unit is 0.5 meter/dose in the figures.

Figure 8 shows the navigation simulation result with no obstacle. The robot route was pre-defined as 1→2→3, where the label (1) is the start point (3.6, 7.2), and the label (3) is the destination (5.4, 8.5). The coordinate of label (2), i.e., midway, is (4.7, 2.3). As can be seen, the robot route was very smooth, and the destination point was reached accurately via the midway.

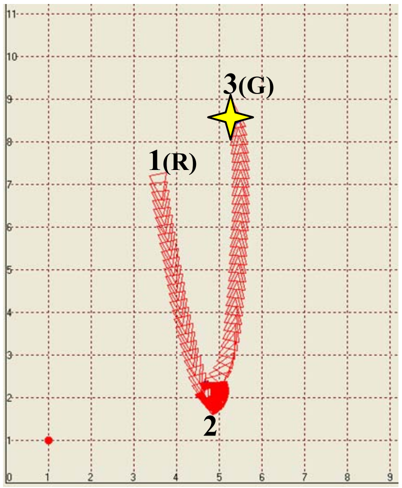

Fig. 8 The navigation route without obstacle

Figure 9 shows the navigation simulation results with adding two obstacles. The coordinates of start point, the destination, the first obstacle, and the second obstacle are set as (5.1, 1.9), (4.6, 8.8), (4.2, 6.9), and (5.6, 4.5). The outcome indicates that the robot can avoid the obstacles and arrive at the destination fast and accurately.

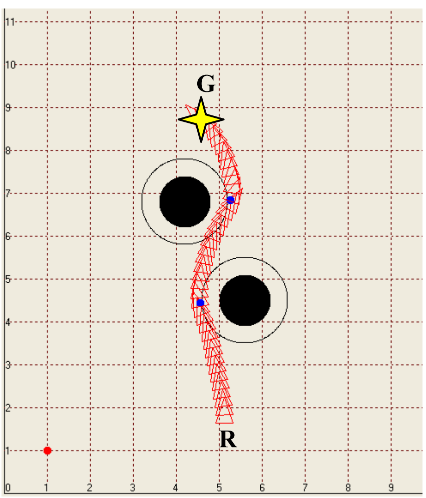

Fig. 9 The navigation route with obstacles

B. Experimental Results with Real Hardware

In the hardware performance, the coordinate unit is 0.5 meter/dose, and the moving speed of robot was set as 0.5m/s. The robot platform to carry out the experimental implementation was designed especially for the meal service robot, shown in Fig.10. This robot's head is equipped with SICK NAV 200 to offer the robot self-locating function. The waist is equipped with SICK LMS291 and the ring ultrasonic sensors for scanning the obstacles.

The practical navigation route with no obstacle is shown in Figure 11. The robot moving route is predefined from 1→2→3. This outcome reveals that the robot route is close to the simulation result (Fig. 8). The localization error is found within ±2cm.

Fig.10 Platform of the mobile robot

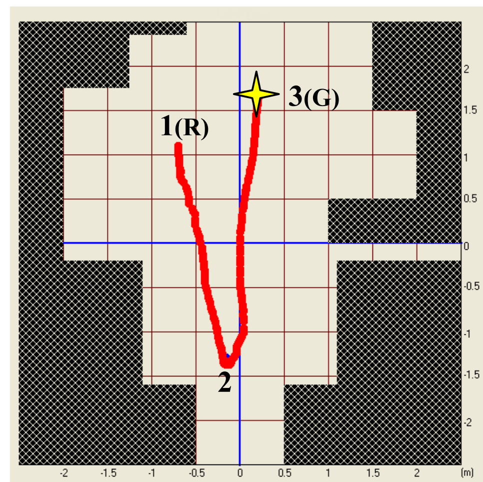

Fig. 11 Practical navigation with no obstacle

The practical navigation result with two obstacles is shown in Fig.12. The near obstacle is firstly found on the way to the destination so that the first midway is determined. The route must be adjusted again when the second obstacle is found on the way, and the second midway is then determined. Obviously, the robot can avoid these obstacles and reach the destination. Actually, the same procedure has been extended to such the case as more obstacles existing. The localization error always does not exceed ±2cm.

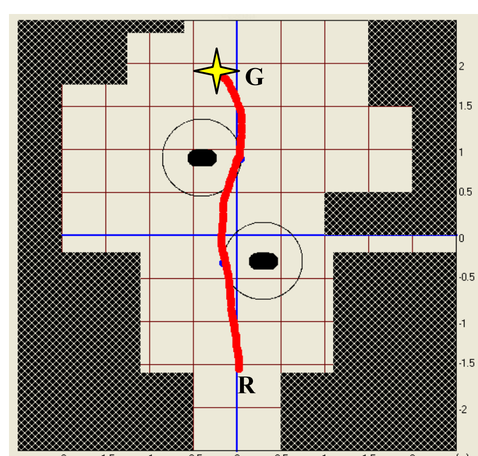

Fig.12 Practical navigation with two obstacles

V. CONCLUSIONS

The highly accurate localization and navigation to mobile robot has been developed successfully in this paper. The Tangent method is used to determine the shortest moving route. The Kalman Filtering algorithm is then applied to amend the route at the k^{th} time, and the route for next $(k+1)^{th}$ time can be estimated. Additionally, the robot route is adjusted to be smooth by the fuzzy logic controller. Accordingly, the robot can reach the destination fast and smoothly within no exceeding ±2cm localization error even there may be several obstacles on the way. Both simulation and practical results confirm that the proposed scheme can be applied to the mobile robots in term of fast and accurate response to achieve the optimal route.

ACKNOWLEDGEMENTS

This research was supported by National Science Council under grant NSC 97-2221-E-167-018- MY3.

REFERENCES

[1] Menegatti E., Pretto A., Scarpa A., and Pagello E. "Omnidirectional Vision Scan Matching for Robot Localization in Dynamic Environments," Vol. 22, Issue 3, June 2006, pp.523-535.

[2] Cang Y., "Navigating a Mobile Robot by a Traversability Field Histogram," IEEE Transactions on Systems, Man, and Cybernetics, Part B, Vol. 37, Issue 2, April 2007, pp.361-372.

[3] Guo-Shing Huang, and Jie-Cong Ciou, "Optimal Route Planning Algorithm of Mobile Robot," To be published on Far East Journal of Experimental and Theoretical Artificial Intelligence, March 2009.

[4] Ouadah N., Ourak L., and Hamerlain M., "Implementation of an Oriented Positioning on a Car-Like Mobile Robot by Fuzzy Control," IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference on Nov. 2006, pp.4076-4081.

[5] Guo-Shing Huang, Jie-Cong Ciou, "Optimal Route Planning Algorithm of Mobile Robot," To be published on Far East Journal of Experimental and Theoretical Artificial Intelligence, Oct. 2008.

[6] Fraichard T., Garnier P., "Fuzzy Control to Drive Carlike Vehicles," Robotics and Autonomous Systems, Vol. 34, No. 1, 2001, pp. 1-22.

[7] M. L. Lee, H. Y. Chung , and F. M. Yu, "Modeling of Hierarchical Fuzzy Systems," Fuzzy Sets and Systems, Vol. 138, No. 2, 2003, pp. 343-361.

[8] W. G. Han, S. M. Baek , and T. Y. Kuc, "Genetic Algorithm Based Path Planning and Dynamic Obstacle Avoidance of Mobile Robot,"1997 IEEE International Conference on Systems, Man, and Cybernetics, Vol. 3, pp. 2747-2751, Oct. 1997.

[9] Guo-Shing Huang, Yan-Hao Chien, "Frequency Stabilizing Using GPS Carrier Phase and Extension Controller," 2006 CACS Automatic Control Conference, pp.1111-1116, St. John's University, Tamsui, Taipei, Taiwan, Nov. 10-11, 2006.

[10] Guo-Shing Huang, Yan-Hao Chien, "GPS/Inertial Sensors Navigation on the Vehicle," AVEC'06 Proceedings of 8th International Symposium on Advanced Vehicle Control, pp.807-812, Taipei, Taiwan, Aug. 20-24, 2006.

[11] Ming-Yuan Shieh, Wei-De Wu, "The Design and Implement of the Robot with the Self-Protection Function," Southern Taiwan University of Technology, Department of Electrical Engineering, Tainan, Taiwan, Nov. 2005.

[12] Scheuer A., Fraichard Th., "Planning Continuous Curvature Paths for Car-like Robots," In Proceedings of the 1996 International Conference on Intelligent Robots and Systems, Vol. 3, Osaka, Japan, pp. 1304-1311.

[13] Li T., Chang S. , Chen Y., "Implementation of Human-like Driving Skills by Autonomous Fuzzy Behavior Control on an FPGA-based Car-like Mobile Robot," IEEE Transactions on Industrial Electronics, Vol. 50, No. 5, Oct. 2005, pp. 867-880.

[14] Ti-Chung L., Chi-Yi T., and Kai-Tai S., "Fast Parking Control of Mobile Robots: A Motion Planning Approach With Experimental Validation," IEEE Transactions on Control Systems Technology, Volume 12, Issue 5, Sep. 2004, pp.661-676.

[15] Chang C., Song K., "Environment Prediction for a Mobile Robot in a Dynamic Environment," IEEE Transactions on Robot Automation, Vol. 13, Dec. 1997, pp. 862-872.

[16] Xiaoyu Y., Moallem M., and Patel R.V, "A Layered Goal-Oriented Fuzzy Motion Planning Strategy for Mobile Robot Navigation," IEEE Transactions on Systems, Man, and Cybernetics, Part B, Volume 35, Issue 6, Dec. 2005, pp.1214-1224.

[17] Gueaieb W., Miah S., "An Intelligent Mobile Robot Navigation Technique Using RFID Technology," IEEE Transactions on Instrumentation and Measurement, Volume 57, Issue 9, Sep. 2008, pp.1908-1917.