International Conference on Control, Automation and Systems 2007
Oct. 17-20, 2007 in COEX, Seoul, Korea

Navigation Strategy for the Service Robot in the Elevator Environment

Jeong-Gwan Kang, Su-Yong An, and Se-Young Oh
LG 302, POSTECH, San 31 Hyoja-dong, Namgu, Pohang, Korea$^{1-2}$, 790-784
(Tel : +82-54-279-2880; E-mail: naroo1, grasshop, syoh@postech.ac.kr)

Abstract: Recently, various service robot systems are introduced and offer the guidance and delivery service. However, most of them are operated on a single floor, because they do not consider the elevator environment. In this paper, we present the navigation strategy for multi-floor environment including elevator situation. Firstly, we propose the recognition method of the button and the floor number so that the robot reacts flexibly to a current elevator status. We apply adaptive threshold method to a current image in order to get a binary image and make the candidates of the button and the floor number using additional filtering method. By rejecting weak candidates using the artificial neural network, we finally get the position of button and current floor number. Secondly, we suggest the path planning algorithm for navigating in the elevator. By constructing an occupancy grid map of the inner part of the elevator, we calculate the optimal position to move and make a shortest path using the distance transform.

Keywords: mobile robot**,** robot navigation, distance transform, path planning

1. INTRODUCTION

Over the past years, the numbers of robots that have been deployed in museums, trade shows and exhibitions has been grown steadily and have offered much kind of services (guidance, delivery, cleaning, patrol, etc.) with high degrees of reliability.

Most services that the robot offers are based on the autonomous navigation. So, the autonomous navigation is one of the most important fields in the intelligent mobile robot.

The representative intelligent service robot is Minerva [1]. The Minerva were deployed in a museum during two weeks and offered the guidance service.

Xavier [2] is the web-based autonomous robot. It estimates its location with the POMDP based localization algorithm and avoids the obstacle using the Lane-Curvature Method.

The robot HERMES [3] has been demonstrated in a long-term test in a museum (both office and exhibition area) where it interacted and communicated with staff and visitors and actually performed useful services several hours a day for six months. In addition, many other service robots were introduced and they performed the autonomous navigation successfully in the museum or office environment. However, their autonomous navigation capabilities consider only the single-floor environment. So they are not able to navigate to the goal on another floor.

To navigate in multi-floor environment, mobile robot should use an elevator. As a first step for moving to the other floor, the robot recognizes the up/down button of the elevator and presses it for calling. After calling the elevator, the robot continuously checks the indicator, which exists above the elevator door and shows a current moving direction of the elevator. If the moving direction of the elevator coincides with that of the robot, the robot gets in the elevator when the elevator door is opened.

In the elevator, as second step, the robot stops in front of the operating panel and then presses the button of the goal floor. Next, the robot moves to the center of the elevator and checks the current floor from indicator. Finally, if the elevator stops at the goal floor, the robot gets off the elevator.

The GRACE [4] is able to detect and getting on/off the elevator using the laser scanner range sensor. However the GRACE cannot recognize where the elevator stops, because it does not have the number recognition algorithm.

The MOPS [5] detects the state of the elevator doors using the LIDAR sensors and the manipulation of the elevator (selecting the floor, getting the floor information, etc.) is realized with the infrared communication system. Because of this, the MOPS is not able to get on/off the elevator which does not have the infrared communication system.

For the robot using the elevator without other help, we first suggest the recognition method of the elevator status including the button, current floor and the moving direction. Then, we propose the navigation strategy especially path planning that enables the robot to get on/off the elevator safely.

This paper is structured as follows. The next section present about the image processing algorithm for recognizing the button and number of the elevator and the third section presents the getting on/off method and localization method in the elevator. The last section presents the simulation and experiment results.

2. BUTTON AND NUMBER RECOGNITION

For getting on/off the elevator, the robot should recognize the elevator status, which is consists of the button, current floor and the moving direction. Simply, let us call all these recognition targets ‘object’.

The four different objects that robot has to recognize are:

- The up/down button at the outside of the elevator.
- The current moving direction at the outside of the elevator.

978-89-950038-6-2-98560/07/$15 ⓒICROS

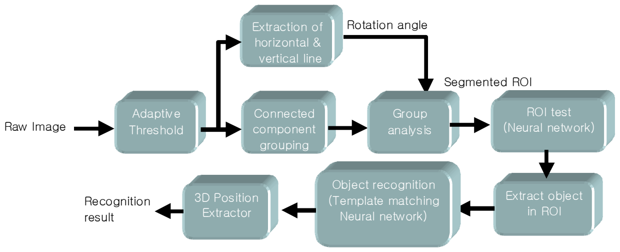
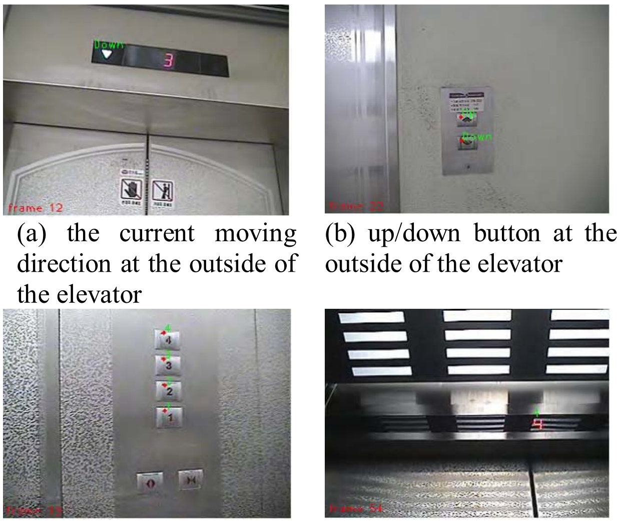

Raw Image
Adaptive Threshold
Extraction of horizontal & vertical line
Rotation angle
Connected component grouping
Group analysis
Segmented ROI
ROI test (Neural network)
Extract object in ROI
Object recognition (Template matching Neural network)
3D Position Extractor
Recognition result

Fig. 1 The block diagram of the button and floor number recognition


(a) the current moving direction at the outside of the elevator

(b) up/down button at the outside of the elevator

(c) the floor button at the inside of the elevator

(d) the current floor at the inside of the elevator

Fig. 2 The button and number recognition results

- The floor button at the inside of the elevator.
- The current floor at the inside of the elevator.

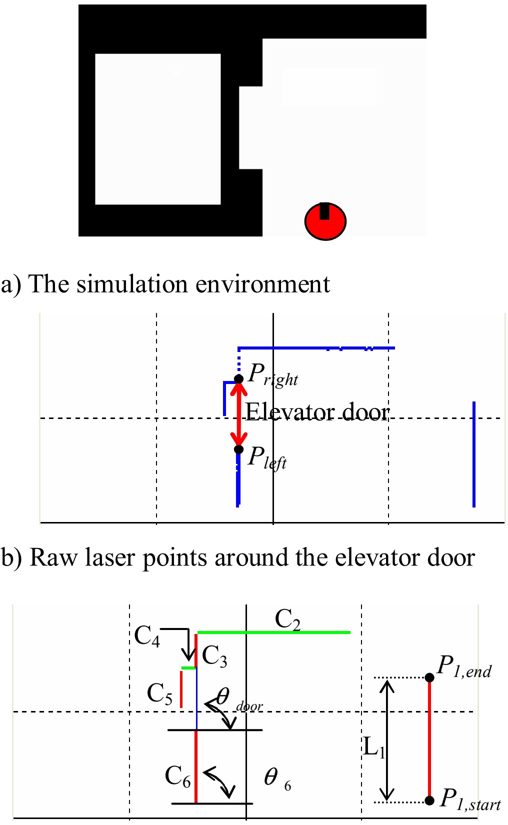
In this paper, we propose the object recognition algorithm based on the adaptive threshold method and artificial neural network. The overall block diagram is shown in Fig. 1.

2.1 Extraction of Object Candidate

All four objects get high pixel value in R channel of the image; therefore, pixel values of G and B channels are less important, so that we do not have to inspect all RGB value to recognize objects. By applying adaptive threshold to R channel of the color image and connected component grouping, we can extract the potential objects while removing non-objects and noise. In the next step, size filtering [8] is applied then we obtain reduced number of object candidates. Size filtering includes width to height ratio and minimum area test.

2.2 Object Recognition

The objects of non-interest are rejected through neural network classifier, which is pre-trained by supervised learning. Training data consists of various object/non-object images. Template matching is applied to object candidate which satisfies neural network classifier. If the correlation between object candidate and template is lower than certain threshold value, then the object candidate is rejected as non-object. Correlation is given by



(a) The simulation environment

(b) Raw laser points around the elevator door

(c) Clustering laser points

Fig. 3 Detecting elevator door using laser scanner

$$R(x,y)=\frac{\sum_{x'}\sum_{y'}T(x',y')\cdot I(x+x',y+y')}{\sqrt{\sum_{x'}\sum_{y'}T(x',y')^2}\cdot\sqrt{\sum_{x'}\sum_{y'}I(x+x',y+y')^2}} \quad (1)$$

, where T and I are template and input image respectively. If same objects are recognized in the stereo image, then 3D position of recognized object is calculated using current robot position and object coordinate in image frame.

3. LOCALIZATION AND MAP BUILDING

We make an occupancy grid map of the the elevator using laser scanner. In order to make the grid map, the robot should know its own position and it means that the map building also includes a localization problem. In this paper, the localization problem is solved using dead reckoning improved by an Extended Kalman Filter (EKF).

3.1 How to detect the elevator door

The robot is positioned as shown in Fig 3(a), the system will see laser readings like those in Fig 3(b).

For finding the position of the elevator door, P_{right} and

P$_{left}$, the algorithm that we developed is as follows :

- Clustering the laser scan points
- Finding the clusters which is longer than 0.5m
- Finding the cluster pairs, C_i and C_j. The distance between end point of cluster C_i and start point C_j is within 1.4 ~ 1.6m
- Finding the cluster pairs. C_i and C_j. The absolute value of the angle differences, $\left|\theta_i - \theta_j\right|$, $\left|\theta_i - \theta_{door}\right|$, $\left|\theta_j - \theta_{door}\right|$, have to be lower than specific threshold
- Finding the position of the elevator. The end point of cluster, C_i ,is to be the right boundary of the elevator, P_{right} and the start point of cluster, C_j, is to be the left boundary of the elevator

3.2 Mobile robot localization using EKF

We calculate the position of the robot using deadreckoning improved by EKF on the assumption that the robot has a map of the working environment and knows the position of the elevator door and wall.

For compensating the deadreckoning error of the robot, the EKF is divided two steps: prediction step and correction step.

In prediction step, the system estimates the position of the robot from the odometry data.

In table 1, $\overline{\mu}_t$ is the position of the robot based on robot's odometry and is calculated as follows:

Table 1. The Extended Kalman filter algorithm

```
1  Algorithm EKF( $\mu_{t-1}$, $\Sigma_{t-1}$, $u_t$, $z_t$ ):
2  $\overline{\mu}_t = g(u_t, \mu_{t-1})$
3  $\overline{\Sigma}_t = G\Sigma_{t-1}G_t^T + R_t$
4  $K_t = \overline{\Sigma}_t H_t^T (H_t \overline{\Sigma} H_t^T + Q_t)^{-1}$
5  $\mu_t = \overline{\mu}_t + K_t(z_t - h(\overline{\mu}_t))$
6  $\Sigma_t = (I - K_t H_t)\overline{\Sigma}_t$
7  Return $\mu_t$ , $\Sigma_t$
```

$$g(u_t, \mu_{t-1}) = \mu_{t-1} + \begin{bmatrix} -\frac{\hat{v}_t}{\hat{\omega}_t}\sin\theta + \frac{\hat{v}_t}{\hat{\omega}_t}\sin(\theta + \hat{\omega}_t \Delta t) \\ \frac{\hat{v}_t}{\hat{\omega}_t}\cos\theta - \frac{\hat{v}_t}{\hat{\omega}_t}\cos(\theta + \hat{\omega}_t \Delta t) \\ \hat{\omega}_t \Delta t \end{bmatrix} \qquad (2)$$

The matrix G$_t$ is the jacobian matrix of $g(u_t, \mu_{t-1})$ and the matrix R is the additional motion noise.

$$G = \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} = \begin{bmatrix} 1 & 0 & -\frac{v_t}{\omega_t}\cos + \frac{v_t}{\omega_t}\cos(\theta + \Delta\theta) \\ 0 & 1 & -\frac{v_t}{\omega_t}\sin + \frac{v_t}{\omega_t}\sin(\theta + \Delta\theta) \\ 0 & 0 & 1 \end{bmatrix} \qquad (3)$$

$$where,\ v_t = \frac{v_l + v_r}{2},\ \omega_t = \frac{v_r - v_l}{2}$$

Using the G$_t$ and R$_t$, we can predict the covariance matrix $\overline{\Sigma}_t$, which represent the uncertainty of the robot's position.

In correction step, we estimate the position of the feature points based on the predicted robot position and compare it to the feature points which are measured from the sensor.

When the predicted robot position is $\overline{\mu}_t$, the estimated feature position is calculated as follows:

$$h(\overline{\mu}_t) = \hat{z}_t = \begin{bmatrix} \sqrt{(m_x - \overline{\mu}_{t,x})^2 + (m_y - \overline{\mu}_{t,y})^2} \\ \tan^{-1}(\frac{m_y - \overline{\mu}_{t,y}}{m_x - \overline{\mu}_{t,x}}) - \overline{\mu}_{t,\theta} \end{bmatrix} \qquad (4)$$

Where (m_x , m_y) are the coordinates of the feature point detection at time *t*.

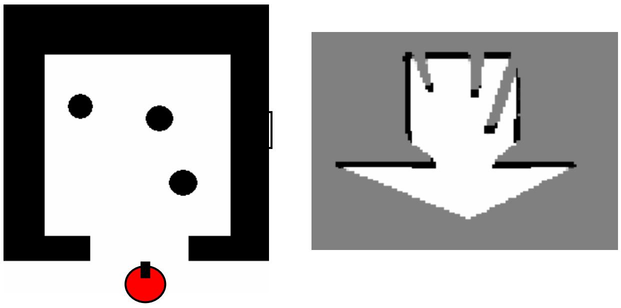
$$H = \frac{\partial h(\overline{\mu}_t, m)}{\partial x_{t-1}} = \begin{bmatrix} -\frac{m_x - \overline{\mu}_{t,x}}{\sqrt{q}} & -\frac{m_y - \overline{\mu}_{t,y}}{\sqrt{q}} & 0 \\ \frac{m_y - \overline{\mu}_{t,y}}{\sqrt{q}} & -\frac{m_x - \overline{\mu}_{t,x}}{\sqrt{q}} & -1 \\ 0 & 0 & 0 \end{bmatrix} \qquad (5)$$

$$where,\ q = (m_x - \overline{\mu}_{t,x})^2 + (m_y - \overline{\mu}_{t,y})^2$$

The matrix H is the jacobian matrix of measurement function $h(\overline{\mu}_t)$, then, the kalman gain K is calculated in line 3. When, the position of feature point which measured from the sensor is given z_t , the robot position and the uncertainty of the robot position are updated in line 5 and 6.

3.3 Map Building

As mentioned above, we are using an occupancy grid map for riding the elevator. In mobile robotics, the occupancy grid is a two-dimensional tessellation of the environment into a grid **G** where each cell C_{ij} holds a part of environment information. The information can be of probabilistic or evidential character and is gathered with sensor (in our case laser scanner) mounted on the mobile robot. In most cases an occupancy grid is created by a mobile robot exploring the environment, updating the grid recursively while moving. Often the updated cell information is just a number that represents the belief that this part of the environment is occupied by an object or free. We use a



(a) Elevator environment (b)Occupancy grid map

Fig. 4 The occupancy grid map in elevator

Bayesian approach in this work for the occupancy grid map building.

The Bayesian approach relies on the Bayes rule:

$$P(A_i \mid B) = \frac{P(B \mid A_i)P(A_i)}{P(B)} \tag{6}$$

For a grid map building purpose the event A_i plays the role of a cell being occupied or not. Each cell C_{ij} of the grid map **G** is therefore associated with a binary random variable S_{ij} with state (O)cupied or (E)mpty for which:

$$P(S_{ij} = O) + P(S_{ij} = E) = 1 \tag{7}$$

Each cell C_{ij} in the Bayesian grid map has a state probability p_{ij} associated and is initialized as follows:

$$p_{ij} := \frac{P(R \mid S_{ij} = O)p_{ij}}{P(R \mid S_{ij} = O)p_{ij} + [1 - P(R \mid S_{ij} = O)][1 - p_{ij}]} \tag{8}$$

where $P(R \mid S_{ij} = O)$ represents the sensor model [7] .

4. PATH PLANNING

4.1 Fitness function

In many cases, there are some passengers in the elevator and their positions can not be estimated before the elevator door open. Because of this, we are not able to decide to the robot's stop position in the elevator.

Thus, for find the stop position in the elevator, we calculate the fitness value from the occupancy grid map. Using Eqs. (9), (10), and (11), we select the optimal position G that has maximum fitness value to get in the elevator.

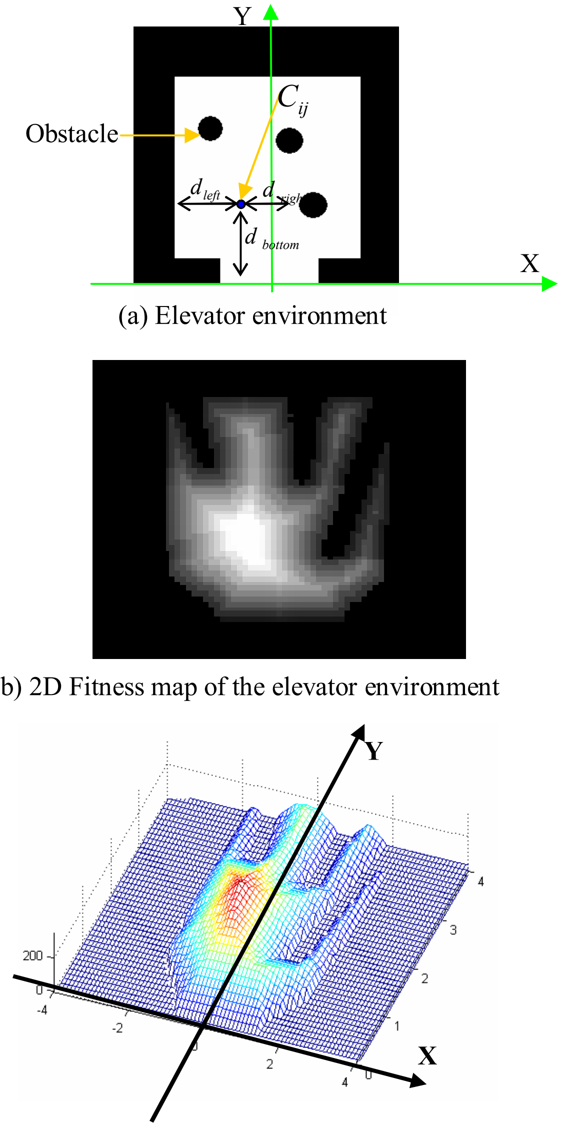
$$d_{\min} = \min\{d_L, d_R, d_B\} \tag{9}$$

$$f(i,j) = K_1 \cdot d_{\min} + d_L + d_R + \frac{K_2}{d_L - d_R + 1} + \frac{K_3}{d_B - offset + 1} \tag{10}$$

$$G = \arg\max_{i,j}\{f(i,j)\} \tag{11}$$

For arbitrary cell C_{ij} in the occupancy grid map (Fig 5(a)), the d_L is the distance from C_{ij} to the obstacle on the left side, the d_R is distance to the obstacle on the right, and d_B is distance to the bottom.

Using Eq (10), we calculate the fitness value of each cell C_{ij}. The cell C_{ij} has a high fitness value when the distance from C_{ij} to other object is large and the distance to the elevator door is close to offset value. So the cell G in Eq(11) corresponds to optimal position for robot stopping in the elevator and Fig 5 shows the fitness value for occupancy grid map of the elevator.



(a) Elevator environment

(b) 2D Fitness map of the elevator environment

(c) 2D Fitness map of the elevator environment

Fig. 5 Calculation of the fitness in the grid map

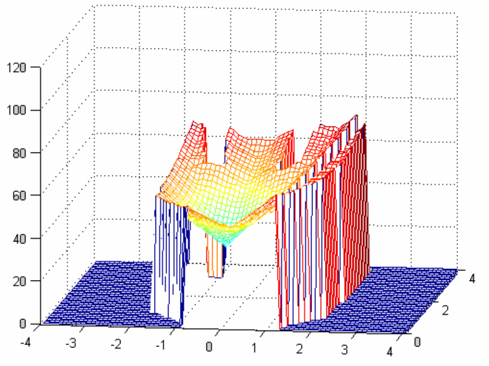


Fig. 6 Distance transform result on elevator map

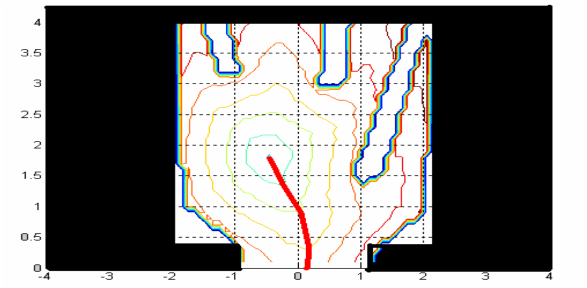


Fig. 7 Collision free path for getting in the elevator

4.2 Path Planning

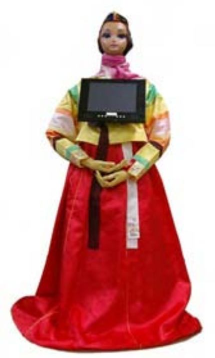
To move to the goal which has the maximum fitness value in the fitness map, we make an optimal path using distance transform. The path planner considers the task of path planning to find paths from the goal location back to the start location. Fig 6 shows the distance transform results. Thus, for any starting point within the environment representing the initial position of the mobile robot, the shortest path to the goal is traced by walking down hill via the steepest descent path and Fig 7 represents the optimal path to goal.

5. EXPERIMENTAL RESULT

The proposed method has been implemented on our mobile robot shown in Table 2. The robot used in experiment is an ActiveMedia PioneerDX2 mobile robot equipped with a HOKUYO URG-04LX laser range finder and two web cameras. For making human like appearance of the robot, we mounted a mannequin and put the clothes on.

The experiment was carried out for several simulated and real environments. Fig 8 shows the robot's trajectory when the robot gets into the elevator in simulator environment. The robot can ride the elevator independent of the robot's start position and the obstacle position in the elevator. (Fig 8(a), Fig 8(b)). Especially.

Table 2 Pioneer DX2 mobile robot used in experiments

	•MORIS (MObile Robot for Intelligent Service) •HOKUYO URG-04LX laser range finder for navigation •Two web cameras for the number recognition

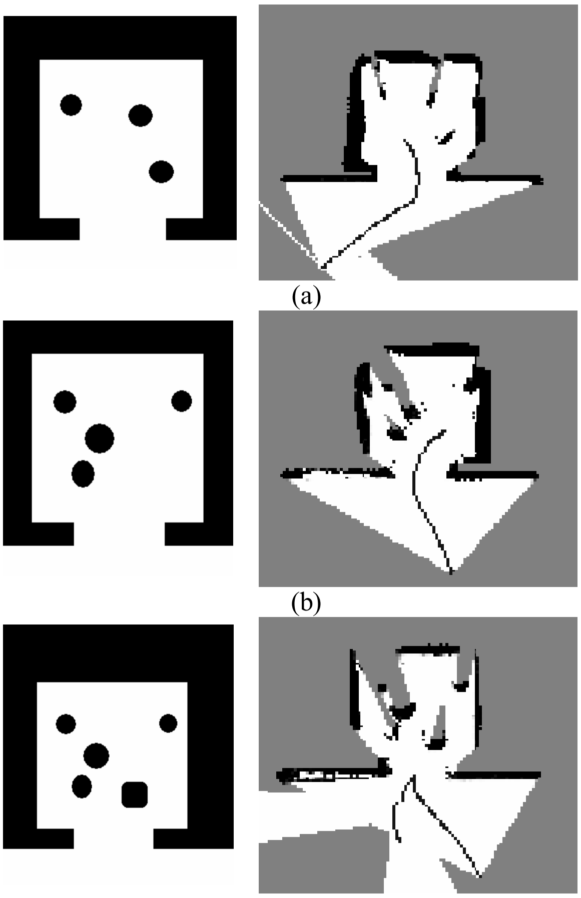

(a)
(b)
(c)

Fig. 8 Collision free path for getting in the elevator

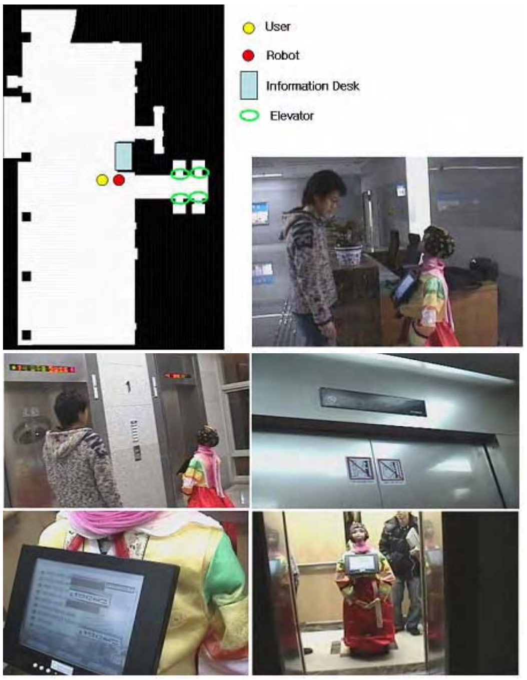


Fig. 9 Experiment results in the Pohang City Hall

Fig 8(c) represent that the robot is not able to get in the elevator because of the obstacle. In this situation, the robot recognizes that getting in the elevator is impossible and return to the start position for waiting for the next elevator.

Fig 9 shows that the robot gets in the elevator safe and moves to the goal floor in the real environment.

6. CONCLUSION

In this paper, we propose the algorithm which makes the robot get in the elevator and move to the goal floor.

First, the robot can check the state of the elevator using the number recognition algorithm based on adaptive threshold, thus our robot can decide get in or not, when the elevator door open.

Second, we make the occupancy grid map about the inside of the elevator in real time. Using this occupancy grid map, we calculate the fitness for robot stop in the elevator and find optimal position for stop. Then, we calculate the path to the maximum fitness position.

However, our robot does not have an arm, Because of this, the robot can not push the button for calling the elevator and can not use the elevator by itself.

For the robot use the elevator without any help, we need more experiment about the pushing the elevator button using the robot arm.

ACKNOWLEDGEMENT

This work was supported by grant No. RTI04-02-06 from the Regional Technology Innovation Program of the Ministry of Commerce, Industry and Energy (MOCIE)

REFERENCES

[1] S. Thrun, M. Bennewitz, W.Burgard, A.B. Cremers, F.Dellaert, and D. Fox, "MINERVA: A Second-Generation Museum Tour-Guide Robot," *Proc. Of IEEE Int. Conf. Robotics and Automation*, pp. 1999-2005, 1999.

[2] R. Simmons, J. Fernandez, R. Goodwin, S. Koenig, and J. O'Sullivan, "Xavier: An Autonomous Mobile Robot on the Web," *Robotics and Automation Magazine*, 1999.

[3] Rainer Bischoff and Volker Graefe, "HERMES – a Versatile Personal Robotic Assistant," Proceedings of the IEEE, Vol. 92, No. 11, pp. 1759- 1779, 2004.

[4] Reid Simmons, Dani Goldberg, Adam Goode, Michael Montemerlo, Nicholas Roy, Brennan Sellner, Chris Urmson, Alan Schultz, Myriam Abramson, William Adams, Amin Atrash, Magda Bugajska, Michael Coblenz, Matt MacMahon, Dennis Perzanowski, Ian Horswill, Robert Zubek, David Kortenkamp, Bryn Wolfe, Tod Milam, and Bruce Maxwell, "Grace: An autonomous robot for the AAAI robot Challenge," *AAAI Magazine*, vol. 42, No. 2, pp 51–72, 2003.

[5] N. Tschichold-Gürman and S.J. Vestli, G. Schweitzer, "The service robot MOPS: First operating experiences," *Robotics and Autonomous Systems*, vol. 34, pp. 165–173, 2001.

[6] Teuvo Kohonen, "The Self-Organizing Map," *Proceedings of the IEEE,* Vol. 78, No. 9, pp. 1464-1480, 1990.

[7] A.Elfes, "Using Occupancy Grids for Mobile Robot Perception and Navigation," *Computer*, Vol. 22, No. 6, 46-57, 1989.

[8] Masahiro Tomono and Shin'ich Yuta, "Mobile Robot Navigation in Indoor Environments using Object and Character Recognition," *Proc. Of IEEE Int. Conf. Robotics and Automation*, pp 313-320, 2000.

[9] Se-Jin Lee, Jong-Hwan Lim and Dong-Woo Cho, "EKF Localization and Mapping by Using Consistent Sonar Feature with Given Minimum Landmarks," *SICE-ICASE International Joint Conference*, pp 2606-2611, 2006.

[10] Sebastian Thrun, Wolfram Burgard, and Dieter Fox, *Probabilistic ROBOTICS*, MIT Press, Cambridge, 2005.