A Network-based Stationeries Rental Service Performed by an Autonomous Mobile Robot

Seydou SOUMARE*, Akihisa OHYA** and Shin'ichi YUTA*
*Intelligent Robot Laboratory, University of Tsukuba
**PRESTO, JST / University of Tsukuba
1-1-1 Tennodai, Tsukuba, 305-8573 JAPAN
{soumare,ohya,yuta}@roboken.esys.tsukuba.ac.jp

Abstract

In this article, the authors report on a practical application of mobile robot playing an active part in human daily life. The application we propose in this study, is about a mobile robot operating in an office environment. The robot interacts and takes frequent orders from humans via a computer network and then navigates through a real world environment to provide them stationeries as items to rent and later on to return back. We present the details of our mobile robot application, our approach and its implementation and, yet show some experimental results.

1 Introduction

For the past years, numerous applications of mobile robot have been developed and proposed. Most of them were made and served mainly for industrial purposes. Only a few mobile robot applications were designed and made with the purpose of supporting humans meaningfully or cooperating with them in order to accomplish a specific task in their daily life.

The few applications of mobile robot designed with a specific and concise task to perform in populated environments, can be grouped in delivery robots [1, 2], guiding robots [2, 5, 6], entertainment robots [3] and cleaning robots [4]. While dealing with real world problems –detailed in [7]–, such as robot navigation in "peopled" environments and the critical issue of human-robot communication and cooperation, these applications were welcomed by researchers and attracted the general public interest into robots. However, beyond the success of taking robots outside of the walls of laboratories and factories to bring them in more humanly milieu, these applications are far from playing an active and meaningful part in humans daily life. They were mostly designed and developed in order to perform various tasks in populated environments; however "are these mobile robot supporting humans usefully in achieving daily tasks?","are humans dependent on a daily base on these applications?" remain as unanswered questions and doubts in most of the general public mind, as well as in the researchers mind.

All these questions can be resumed as the following one : "besides industrial applications, can mobile robots play an useful and meaningful role in human daily life?". Our study is yielded and driven essentially by such interrogation. In order to find an equivocal answer, we propose in this work, an application of mobile robot designed and elaborated with the up most purpose of aiding skillfully humans into accomplishing a daily specific task, while opening new frontiers and possibilities in the future of mobile robots. The application is about a mobile robot that interacts with humans through a computer network as well as in a real environment, offering them services such as stationeries to rent or return back; the stationeries are possessed and managed by the robot, in its body.

In this paper, we introduce first, the whereabouts of a network-based stationeries rental service performed by an autonomous mobile robot and outline its basic requirements or functions needed in order to be materialized. Next, we present our realization strategy and show our robot system architecture and its implementation. Last, we try to bring an answer to the main question stated above, thanks to some experimental results.

2 What is a Network-based Stationeries Rental Service ?

2.1 Description

In this study, we define a network-based stationeries rental service performed by an autonomous mobile robot as the following task.

1. A person, let's name X, specifies from the screen of her computer, a stationery as an item she immediately needs for performing or finishing tasks in an office environment.

2. From a remote location, a mobile robot tooled with various stationeries inside a box set on the top of its body, gets that information through a communication media.

3. Then, the robot checks in real time if it has the stationery sought by X inside the item box.

4. If "Yes", then the robot moves through the office till X location and rents the item to X.

5. If the sought item is not present at that time inside the box, which means that another person let's name Y, has rented it, in a previous action, the robot contacts Y through her computer screen inquiring if she is ready to return the specific item back to the robot.

6. Based on Y's answer, the robot either moves to Y location to get the item back and then visits X to rent her the item, or the robot just transmits to X that the item rental service can not be provided due to someone else still using the item sought.

7. In addition of the above tasks, the mobile robot also furnishes to inhabitants of the office environment updated informations of all items rental status and delivers desired stationeries at an appointed time to humans who previously booked items for later use.

2.2 Needed Functions

The tasks defined in the previous section bear a variety of difficulties which have to be solved in order to realize this robot application. Here, we outline the basic and needed functions that a mobile robot must fulfill in order to perform with a minimum of dexterity the task described above.

Human Interface : an interface, shaping mutual communication, occasionally needed cooperation and sufficient understanding between humans and the robot system, is one of the primordial functions needed in this application. This interface should enable for instance distant humans to access easily to the robot system for renting items. In addition, the robot should show to its human clients the location of the sought item inside the box, through a specific medium, for the physical handling of items.

Autonomous Navigation : since it is operating in -heavily- populated environments like offices, the robot should be able to navigate optimally between multiple locations inside the office, percept and avoid static, as well as dynamic obstacles, localize frequently itself to avoid misplacement errors.

Item Perception and Management : as this application is about moving between different humans locations to rent or get back stationeries, the robot has to detect autonomously items present in the box set on the top of its body. It must also, have little understanding of the locations of all rented items and coordinate sequentially its behavior toward human requests, based on the rental state of items.

3 Our Approach

In this section, we discuss about the strategy thought for realizing this application and also explain our robot system hardware and software architecture.

3.1 Strategy

Below, we define our strategy accordingly to the above three basic functions needed to realize this robot application.

Concerning the first issue of human-robot interaction, we first equip our mobile robot with a portable personal computer connected to a computer network via a wireless Ethernet card. Doing so simplifies greatly the mutual communication needed between the robot system and any distant human using a desktop PC, as a simple network communication protocol between computers. Consequently, we have to develop, for human purpose, a graphical user interface maneuverable by a pointer device(mouse). Another human interface developed in this study, is a speech and visual effect interface through which the mobile robot shows the location of sought items, when it delivers them to humans. We will discuss further details of our human-robot interaction implementation in the next section.

About the problem of navigating in populated places as an office environment, we use an existing navigation software system "MaP / RouteRunner", to be explained later on this paper.

The item perception and management is a functional property to this application besides interacting with humans and moving through their locations. Therefore, in this study we select a variety of everyday use stationeries and develop sensors appropriate to each of them; doing so enable our mobile robot to percept their existence, once items are placed inside the item box. Also beyond detecting stationeries presence in the box, the robot manages autonomously all items, which means here to memorize in a database the whereabouts of every item after a delivery service.

3.2 Hardware Architecture

Our hardware architecture consists simply of a mobile robot, an item box and a portable personal computer connected via the computer network to desktop PCs used by humans, as illustrated in Figure 1.

First, the robot that carries out the task is based on the mobile robot platform "Yamabico", which has been developed by our research group. Yamabico is about 50cm(W) x 50cm(D) x 60cm(H) of size and is a two-driving wheeled self-contained robot with an odometer. It is equipped with a bumper panel touch sensor in its front and four ultrasonic sensors in four directions, to monitor the environment. The odometer is the primary sensor during navigation, while ultrasonic sensors serve with the odometer for self-positioning, and the bumper panel enable to prevent

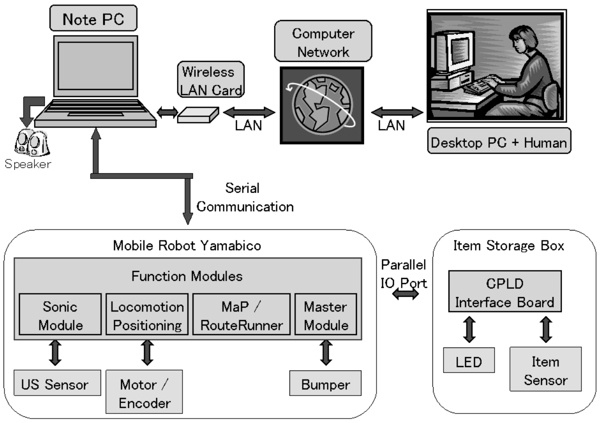


Figure 1: Hardware components

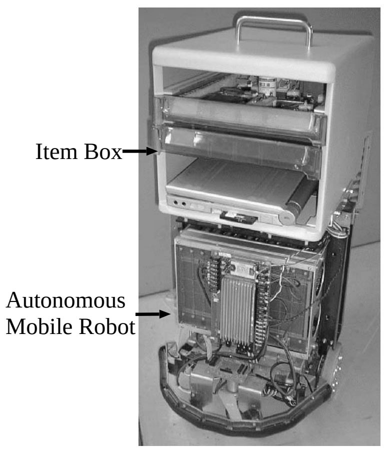


Figure 2: Autonomous mobile robot "Yamabico" supporting the item box

harm in emergencies. Yamabico's control system consists of several modules, as a vehicle control module, an ultrasonic sensor control module, a self-positioning module and a master module. The latter module controls the robot behavior by maneuvering other modules via an inter-module messaging communication. Details on the Yamabico robot architecture and vehicle control are presented in [8, 9, 10].

On the top of Yamabico, we set an item box as shown in Figure 2, in which are set items combined with their respective sensors and light-emitting diodes used for visual effect. Next, we equip the overall robot system with a notebook PC and a wireless Ethernet card. The notebook PC is connected to the Yamabico robot via a serial port communication device.

3.3 Software Architecture

Our robot system has a bottom-up software architecture. As we illustrated in Figure 3, the bottom-up architecture consists of three layers : navigation layer, task manager layer and user interface layer. These layers are implemented as separate code modules(processes). On one hand, the navigation layer lies inside the Yamabico's master module, which commands vehicle control, sonar environment perception among others modules. The task manager and user interface layers reside within the portable computer. The interface within layers is operated by an asynchronous inter-process messaging. Below, we explain the details and functionality of each layer.

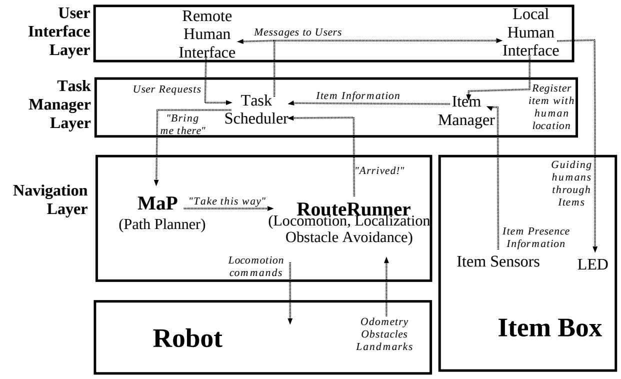


Figure 3: Software architecture

The **navigation layer** is built on MaP / RouteRunner, an already developed and proposed navigation software system in indoor environments for mobile robot purposes. MaP / RouteRunner is divided into two sub-layers which are MaP -the path planner- and RouteRunner which handles functions as path locomotion, localization and obstacle avoidance.

The **task manager layer** plays the decision-maker of our autonomous robot system. It is made of mainly two parts, a task scheduler and an item manager. First, the item manager is the process gathering in real-time stationaries presence in the item box and providing these informations to the task scheduler, when needed. Further, its stores in a database, all items whereabouts. On the other hand, the task scheduler's functionality is basically, to retrieve from humans requests, their location, desired items and service. And, from these informations, the task scheduler verifies the availability of requested items, then appreciates the feasibility of the requested service and lastly transmits its decision to others layers. This is, to send sequentially a destination (i.e location of a human), in case of a feasible request to the navigation layer, and throughout the user interface layer, it messages humans about the finality of their requests. The task scheduler, also inquires distant humans when an item they hold, is needed by others. In addition, the task scheduler receives reports from the lower layer after a navigation task; those informations will be used promptly to activate the item delivery interface, as we explain next.

The **user interface layer** manages the overall interaction between the robot and humans. It is made of a remote

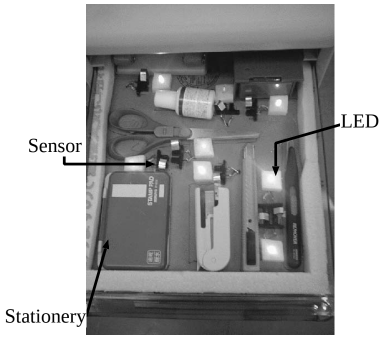
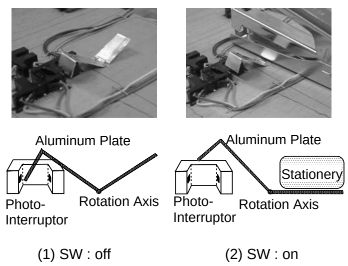
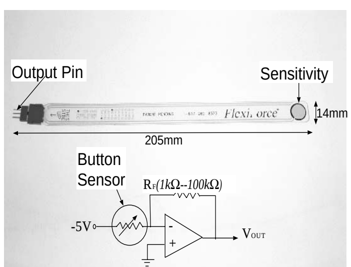


Figure 4: View of stationeries stored inside a drawer of the item box



(a) Photo-Interruptor Sensor



(b) Button Sensor

Figure 5: Stationeries sensors

human interface and a local human interface, implemented as two independent processes. The remote human interface receives, in one direction, requests from humans through the computer network and in the other direction, it sends out messages and inquiries, generated by the task manager layer to different humans through the same network. The local human interface is a speech and visual effect tool the robot uses to guide humans when they are renting or returning back items inside the box.

4 Implementation

4.1 Stationeries Perception

As shown in Figure 4, we dispose a selection of office daily use stationeries in an item box drawers and set the box on the top of our mobile robot body. Inside any given drawer, we place the stationeries within a specific location and immobilized them with some corrugated material.

In order to enable our robot system to percept au-

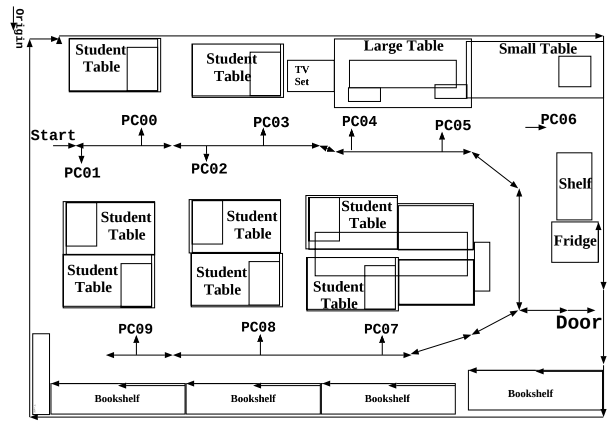


Figure 6: Topological map of a real-world environment provided to the mobile robot

tonomously all stationeries existing inside the item box, we develop appropriate and efficient sensors for percepting stationeries with various shape, size and weight.

One type of stationery sensor, we developed, is a combination of a photo-interruptor sensor and an aluminum plate, as illustrated in Figure 5-(a). The principle of such sensor is simple; when an item is placed on the lower and light side on the aluminum plate, this one tumbles down, hence switching on the photo-interruptor sensor. It is a very simple but effective and low-cost sensor appropriate toward stationeries with a large cross section.

Another type of sensor used in this study, is a touch sensor, "FlexiForce" button sensor, a commercial product of Nitta Corporation. This sensor, as shown in Figure 5-(b), is a force sensing resistor shaped as a thin film. It percepts proportionally to a resistance, the mass of a load in contact with its sensibility part. In our work, we used an analog circuit described in Figure 5-(b), to convert the force sensor resistance into an usable voltage which can be read, above a threshold value, as an item presence information. This sensor is very practical toward detecting heavy and unevenly shaped stationeries.

4.2 MaP / RouterRunner

In this work, we use "MaP / RouteRunner" an already built navigation software system, in order to provide an efficient and reliable navigation function to our robot system. It consists briefly of a path planner –MaP (Map and Planning)–, and a path navigation handler –RouteRunner–.

The path planner searches the optimal shortest route from the robot actual position to a desired heading, based on a topological map of the real-world environment as shown in Figure 6. Figure 6, a sample of topological map provided to the path planner is an usually hand-made map representing all static three-dimensional objects present in the indoor environment. It also includes places information such as the entrance door, humans locations inside the

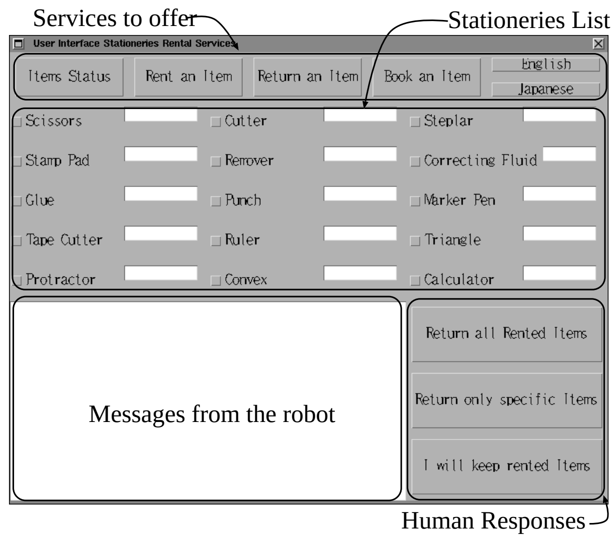


Figure 7: The human interface(GUI) through a computer network

office, which are for example, human real name or aliases, or simply their desktop PC's host name, as illustrated in Figure 6. The overall navigable paths of the robot within the office is shown as a network of navigable segments (a graph). Each navigable segment is priorly defined off line, accordingly the robot width in order to assure a minimum of safety towards sided static obstacles during a navigation action.

After planning the optimal shortest path to a destination, MaP hands it to RouteRunner which, in return, sends locomotion commands to motors, gets odometry data from wheels encoders and landmarks data from sonars, in order to run safely between static obstacles and position itself correctly on the given path till arriving at destination. In case of the robot percepting on the given path, thanks to sonars and the bumper panel, obstacles such as humans or displaced objects(chairs), which are dynamic and unmarked objects in the topological map, RouteRunner stops the navigation action for a while and waits for the dynamic obstacle to move away from the path. If the obstacle still stands on the path, then RouteRunner inquires the path planner for a local avoidance path planning and then, tries the given avoidance path in order to first avoid the obstructing object and then reach the goal destination.

Further details on MaP / RouteRunner functionalities are available in the following references [11, 12].

4.3 Human-Robot Interaction

The backbone of our human-robot mutual communication is built on the Java Remote Methods Invocation(RMI) technology. RMI is a communication system that allows an object running in one computer's Java Virtual Machine, to call and execute methods of an object running in another

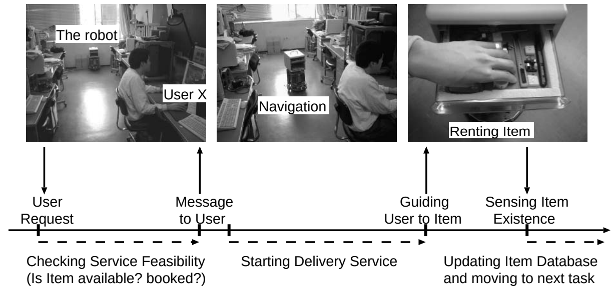


Figure 8: A sequential view of an item rental service to a single user

computer's Java Virtual Machine. RMI provides basic remote and mutual communication through a computer network, between programs written in the Java programming language.

In addition to the communication system, we designed and developed a graphical user interface (GUI) for human purpose. A view of the interface is presented in Figure 7. This interface is the access tool used by distant humans to interact with the autonomous robot system. It consists of three major panels, which are a service panel, a panel listing rental items and a message panel extended with few patterns of answering to the robot inquiries. For humans at any place –any computer– in an office environment, to rent an item using this interface is an easy task going like, "First, check sequentially the items of your preference with the pointer device, then click lastly the desired service and view the robot answering to your request through the messaging panel."

The local human interface, we mentioned in the user interface layer explanation in section 3.3 of this article, is implemented as LED diodes set closely with every stationery and the item box drawers. The interface also includes speech processing via the notebook sound card.

5 Experiments

We realize, in this study, some experiments to evaluate the performance of our system implementation. The experiments were conducted in our laboratory environment, represented in the topological map of Figure 6. The contents of the experiment are resumed as follows.

The primary step of the conducted experiment, is to evaluate the functionality of our system during an item rental service provided to a single user. As illustrated sequentially in Figure 8, a remote human X –located a few meters from the robot–, requests an item (stapler) to the robot, through the GUI (Figure 7). The robot, next appreciates the feasibility of the requested service and then decides autonomously to navigate till X location and delivers the item with the appropriate interfaces.

The next functionality we experimented, is to see if our robot system was able to inquiry efficiently users about

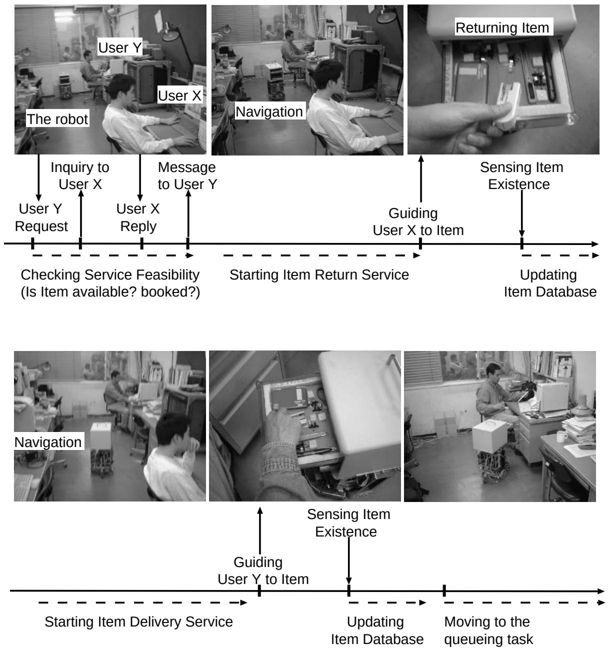


Figure 9: A sequential view of an item inquiry and rental service to different users

rented items in order to get them back, when other users need it or simply, when the time allowed for items rental is over. As illustrated the sequence of images in Figure 9, an user Y, wishes to rent a stationery, previously delivered to X. In this case, the robot invokes user X through the screen of his computer and after getting a favorable answer, it gets back the specific item and delivers it later on to user Y.

Concerning these two steps, our robot performed well in 7 tentatives out of 10. Most failures were due to misperceptions of items, occurring when humans placed them roughly in the box and their respective sensors (photo-interruptor) are not switched off. Another cause of failure is obstacle avoidance, which in our laboratory is optionally limited, a reason why our route planner gives up when all possibles paths are obstructed by dynamic obstacles.

These experiments were conducted by the robot in a limited environment –our laboratory– but they served as significant feedbacks to improve our system human interfaces hardware and software implementation, in order to experiment our application in a real office environment.

6 Conclusions

The purpose of this work is to develop an application of mobile robot interacting with humans through a computer network, as well as in a real-world environment to offer them stationeries rental services. In this article, we presented the background factors yielding to our study, proposed an application of mobile robot taking part actively in human daily life and defined our approach of realizing it. We also outlined, the details of the autonomous robot system we are presently developing, its hardware and software structure. Our next work is to complete the implementation of the system, experiment it in a real-world environment and evaluate its usability by taking in account feedbacks of humans inhabiting the experimental environment.

References

[1] J. M. Evans : “HelpMate : An autonomous mobile robot courier for hospitals”, *Proc. of the International Conference on Intelligent Robots and Systems*, pp.1695-1700, 1994.

[2] R. Simmons, R. Goodwin, K. Z. Haigh, S. Koenig and J. O’Sullivan : “A layered architecture for office delivery robot”, *Proc. of the 1st International Conference on Autonomous Agents*, (Agents ’97 Marina Del Rey, CA), pp.245-252, 1997.

[3] S. W. Park, J. H. Kim, E. H. Kim and J. H. Oh : “Development of a Multi-Agent System for Robot Soccer Game”, *Proc. of the International Conference on Robots and Automation*, pp.628-631, 1997.

[4] E. Prassler, E. Stroulia and M. Strobel : “Office Waste Cleanup : An Application for Service Robots”, *Proc. of the International Conference on Robots and Automation*, pp.1863-1868, 1997.

[5] S. Thurn, M. Bennewitz, W. Burgard, A. B. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Shulte and D. Schulz : “MINERVA : A Second-Generation Museum Tour-Guide Robot”, *Proc of the International Conference on Robots and Automation*, pp.1999-2005, 1999.

[6] S. Maeyama, S. Yuta and A. Harada : “Experiments on a Remote Appreciation Robot in an Art Museum",*Proc. of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.1008-1013, 2000.

[7] S. Thurn : “When Robots Meet People : Research Directions in Mobile Robotics”, *IEEE Intelligent Systems,*May–June 1998.

[8] S. Suzuki, J. Iijima and S. Yuta : “Design and Implementation of an Architecture of Autonomous Mobile Robots for Experimental Researches”, *Proc. of the 6th International Conference on Advanced Robotics*, pp.423-428, 1993.

[9] S. Yuta, S. Suzuki and S. Iida : “Implementation of A Small Size Self-Contained Autonomous Robot - Sensor, Vehicle Control and Description of Sensor Based Behavior -”, R.Chatila et al. Eds, *Experimental Robotics II,*pp.344-359, Springer-Verlag, 1993.

[10] S. Iida and S. Yuta : “Vehicle Command System and Trajectory Control for Autonomous Mobile Robots”, *Proc. of the International Conference on Intelligent Robots and Systems*, pp.212-217, 1991.

[11] T. Ogura and S. Yuta : “Map and Planning Module for Autonomous Mobile Robot”, *Proc. of the 15th Annual Conference of Robotics Society of Japan*, pp.239-240, 1997 (in Japanese).

[12] Y. Hada and S. Yuta : “Robust Navigation and Battery Recharging System for Long Term Activity of Autonomous Mobile Robot”, *Proc. of the 9th International Conference on Advanced Robotics*, pp.297-302, 1999.