Active Navigation Landmarks for a Service Robot in a Home Environment

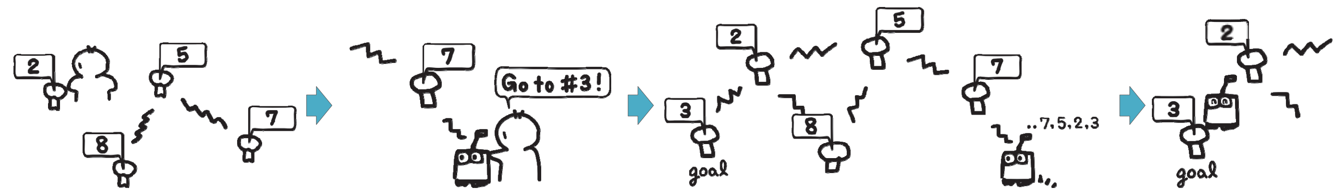
Kentaro Ishii[1], Akihiko Ishida[1], Greg Saul[1], Masahiko Inami[2,1], and Takeo Igarashi[3,1]

[1]Japan Science and Technology Agency, ERATO, IGARASHI Design Interface Project, Tokyo, Japan

[2]Graduate School of Media Design, Keio University, Yokohama, Japan

[3]Graduate School of Information Science and Technology, The University of Tokyo, Tokyo, Japan

{kenta,ishida}@designinterface.jp, gregsaul@gmail.com, inami@designinterface.jp, takeo@acm.org



(a) Placing Active Landmarks (b) Directing the Robot to a Destination (c) Navigation (d) Arriving at the Goal

Figure 1. System overview: (a) The user places the landmarks on the floor. Each landmark has an ID and communicates with the neighboring landmarks. (b) The user directs the robot to a destination by pushing a button on the robot. (c) The robot receives information from the landmarks and navigates by following the landmarks one by one. (d) The robot arrived at the goal.

***Abstract*—This paper proposes a physical user interface for a user to teach a robot to navigate a home environment. The user places small devices containing infrared based communication functionality as landmarks in the environment. The robot follows these landmarks to navigate to a goal landmark. Active landmarks communicate with each other to map their spatial relationships. Our method allows the user to start using the system immediately after placing the landmarks without installing any global position sensing system or prior mapping by the robot.**

Keywords – active landmarks; end-user interface; robot navigation; navigation path; infrared communication

I. INTRODUCTION

This paper addresses the problem of a robot navigating the home environment assisted by an end user. Suppose you want to have a robot be able to move to a specified location in your home without assistances. One way is to first have the robot scout the environment (e.g. Simultaneous Localization and Mapping [1]) and use the acquired information for robot navigation. A problem with this is that automatic scouting can take time, and the user cannot start using the robot immediately. In addition, even after scouting, the robot only has an internal representation of the world. It is not a trivial problem to provide a human usable interface to handle this internal representation (labeling problem). For example, when the user wants to send a robot to the kitchen, the user has to tell the system which location in the surveyed environment is the kitchen. Another approach is to use a global positioning sensor such as ceiling mounted cameras or wireless systems such as Ubisense [2]. However, the installation and calibration are not trivial. The labeling problem also occurs with this global positioning method. Our method is similar to robot localization methods where a robot places communicative devices in the environment in order to map it, for example [3]. In our system, the user maps the environment instead of the robot.

We propose a physical user interface for teaching a robot the map of a home environment without advance scouting or global sensors. Fig. 1 illustrates the basic idea. The user places active landmarks along the path which automatically compute adjacency relationships by communicating with each other. After obtaining the adjacency relationships of the landmarks, the user can send a robot to any landmark by simply specifying the ID of the landmark. The robot computes the shortest path to the target by examining the adjacency relationships and follows the landmarks in the path one by one. This method allows the user to teach a path without prior scouting or global position sensing. In addition, since the active landmarks dynamically calculate the relationships with each other, the user can easily reconfigure the path structure simply by moving the landmarks. This is important because the home environment is not static and objects may have been moved into the robot's path. Our device contains a LED indicator on each communication channel for the user to ensure the landmark can see the neighboring landmark. Thus, the user can adjust the positions of landmarks so that the robot can reliably navigate in the environment. This work aims to make interaction intuitive and smooth by solving the labeling problem as well as to describe implicit human-robot interaction by the user mapping the environment for the robot.

II. DEVICE HARDWARE

We use coded IR (infrared) communication for active landmarks. Each active landmark consists of eight IR LEDs, eight IR receivers, and eight indicator LEDs, two microcontrollers, and battery (Fig. 2). The eight IR LEDs are used for transmitting a signal. It is modulated at 38kHz by one of the microcontrollers. The eight receivers are off-the-shelf modules that receive 38kHz-modulated IR wave, which are usually used in remote control receivers. The eight indicator LEDs are used for user feedback. They indicate the connectivity to the neighboring landmarks, and the user can adjust the positions of the landmarks to ensure the connectivity.

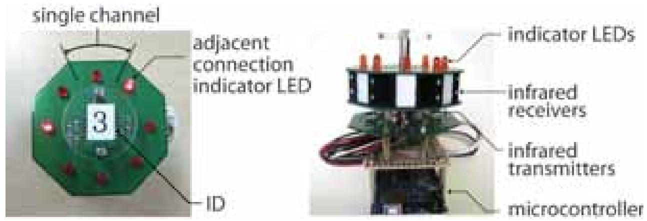


Figure 2. An active landmark consists of eight IR transmitters, eight IR receivers, and eight indicator LEDs, two microcontrollers, and battery. The indicator LEDs display that the adjacent landmarks are detected in that direction.

III. Map Construction

Active landmarks construct an adjacency map automatically by themselves. The user simply places the landmarks along the paths. The adjacency relationships are then computed by the landmarks, which communicate ID and angle information with each other.

Each active landmark stores a table that lists the relative direction of all connected landmarks. Local adjacency information is broadcasted to neighboring landmarks when the information is updated. The receiving landmarks propagate this information to every connected landmark (Fig. 3).

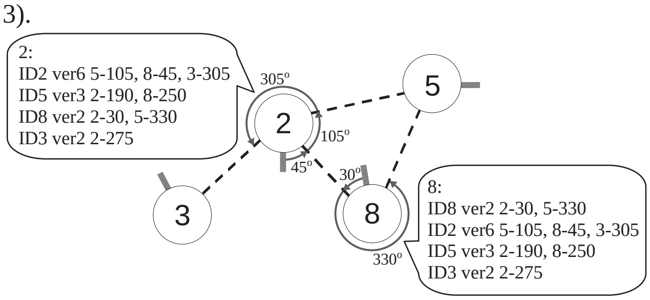


Figure 3. This figure shows the adjacency tables of the landmark 2 and 8 as an example. The active landmarks compute the local adjacency information and propagate it. Each row of the table contains an ID of a landmark, a version number, and pairs of an ID and a direction of its neighboring landmarks.

IV. Robot Navigation

The user needs only to push a button on the robot. Suppose there is the landmark 1 in the kitchen, our method requires the user simply pushes button 1 on the robot when the user want the robot to move to the kitchen. The user does not need to learn the internal representation of the system or to use special interface to handle it. In this way, we solve the labeling problem just by placing landmarks on the floor.

The robot requires IR receivers and a camera for operation with the system. The IR receivers are directed in 8 directions the same as active landmarks as to receive signals from any direction. The camera is used for estimating distance to a landmark.

At each iteration in the program loop, the robot checks the information from the visible landmarks, and moves according to the information. The algorithm for the robot to get to a goal landmark is shown below:

1) The robot receives the adjacency information of all connected landmarks from a landmark near the robot and calculates the path to the goal landmark. If the specified ID of landmark is not connected to the current position, the robot gives feedback by making an alert sound to indicate that it cannot reach the goal.

2) The robot approaches the next landmark based on IR signal from the landmarks. It also looks for the landmark using its camera.

3) If the landmark is detected in the camera's image, the robot estimate the distance to the landmark based on the size of the landmark. After the robot gets close enough to the landmark, it turns to the next landmark.

4) Repeat from step 2 until the robot reaches the goal or the robot cannot find the next landmark. If the robot cannot find the next landmark, the system calls the user by making a sound to let the user rearrange landmarks.

If the active landmarks are properly placed in the environment, the robot can move from any landmark to any other landmark in the connected graph.

V. Current Status

We created the first prototype of the system presented in this paper, and confirmed robot navigation works with our prototype. However, we still have room for improvement. First, distance measurement is done using a camera. We plan to replace the receivers, which only detect an incoming signal, with ones that can measure intensity of the signal to estimate distance. We think that this will make the robot simpler, improve direction accuracy, and increase reliability of the distance estimation. Second, the landmarks may get in the way if it were placed in a frequently traveled area. We plan to redesign the devices to make them smaller and add a housing to make them more durable. Third, battery capacity can also be problematic with the current implementation. Since the active landmarks always run and constantly exchange signals, the battery runs out in 60 hours. According to the specification of the microcontrollers, if they entered into a sleep mode, the battery life could be extended by up to ten times. Applying an algorithm for sleeping in idle times and waking up at a user's trigger of rearrangement or robot navigation as well as replacing the battery and LEDs to ones more efficient will remove this problem. Lastly, our future work includes a usability test of this method by novice users.

References

[1] J. J. Leonard and H. F. Durrant-whyte, "Simultaneous map building and localization for an autonomous mobile robot," Proc. IEEE/RSJ International Workshop on Intelligent Robots and Systems (IROS 91), vol. 3, pp. 1442–1447, 1991.

[2] Ubisense, "Ubisense english site." http://www.ubisense.net

[3] S. Miyama, M. Imai, and Y. Anzai, "Rescue robot under disaster situation: position acquisition with omni-directional sensor," Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 03), vol. 4, pp. 3132–3137, 2003.