2009 International Conference on Adaptive and Intelligent Systems

Environmental modeling and identification based on changes in sensory information

Manabu Gouko and Koji Ito
Department of Computational Intelligence and Systems Science,
Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology,
4259 Nagatsuta, Midori-ku, Yokohama, 226-8502, Japan
Email: gouko@dis.titech.ac.jp

***Abstract*—Adaptability to various environments is needed for a robot that supports our lives. Environmental identification is important for a mobile robot that works in multiple environments (e.g., different rooms). We present an environmental modeling method based on state representation, which represents a change in sensory information. Our model enables the mobile robot to identify which environment it is in. The results of experiments on a real mobile robot with only low-sensitivity infrared sensors show the effectiveness of our method, and a comparison between our method and a conventional one shows that ours has higher performance.**

***Keywords*-environmental modeling; environmental identification; mobile robot;**

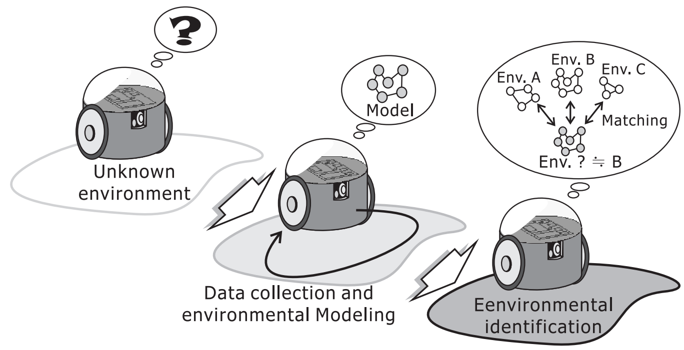

Figure 1. Overview of the environmental identification.

I. Introduction

In research on autonomous mobile robots, many studies on localization and navigation tasks have been conducted. These tasks are generally performed using prior knowledge (e.g., a map of the environment) based on visual and position data of its environment [1], [2], [3], [4]. For example, a robot that works in several environments, such as different rooms, needs knowledge about each environment. If the robot can recognize that it is in a known environment, it can use the map of the surroundings. On the other hand, if the robot recognizes that it is in an unknown environment, it will try to build a new map of the environment. Thus, it is important for a robot to identify which environment it is currently in so that it can work efficiently. Autonomous mobile robots should be able to identify their environments by themselves.

An easy method of identifying multiple environments is to set landmarks in each environment. The robot can identify which environment it is in by comparing the landmarks; however, this requires space and time for installation. Another method is to use environmental modeling and comparison (Figure 1). The robot makes a model of its current environment and compares it against many stored environmental models. For highly accurate recognition, this method requires a model that can represent the characteristics of a real environment. Moreover, the model must be easier to make than a new map.

Research on environmental modeling using stochastic models is well known. Models built using such an approach are applicable not only to environmental discernment but also to localization and navigation. However, building a stochastic model requires a lot of data, which is collected by making the robot move about in the environment. Therefore, it is difficult to use this approach to enable a real robot to make multiple environmental models.

The characteristics (e.g., size and layout) of each environment appear as the form of the environment if the environment is a closed region surrounded by walls. On the assumption that the environment is a closed region, several studies on environmental modeling and identification have been done[1], [5], [6].

Matarić proposed a modeling method for integrating two sets of information from sonar sensors and from a compass[1]. She defined several features about the form of the walls using sensory information and then made a model using those features called landmarks. The model was represented by a graph with nodes corresponding to the landmarks. In that study, the effectiveness of the method was evaluated by using the model for path planning. The landmark-based model is robust against sensor noise, but, in this method, a designer must make the raw sensory data correspond to the landmarks prior to the modeling.

Nehmzow et al. proposed an environmental modeling method using sensory data acquired by making the mobile robot perform a wall-following movement[5]. They used information about the kind the corner (e.g., convex or concave) and the distances between two adjoining corners as environmental characteristics. They input these features into a neural network as input vectors, and the learned network

978-0-7695-3827-3/09 $25.00 © 2009 IEEE
DOI 10.1109/ICAIS.2009.22

could then identify the forms of the environment. However, this work treated only simple environments with right-angled corners and straight walls.

Yamada and Murota proposed a modeling method and tested multiple environment recognition by a real mobile robot[6]. In that method, the robot initially performs a wall-following movement in its environment under the control of if-then rules. Then, the rule sequence observed during wall-following is used for modeling. If environmental models are made using if-then rules instead of using sensor data directly, the acquired models are robust against sensory noise. They confirmed the effectiveness of their method through an experiment with a real mobile robot that had only low-sensitivity infrared sensors. By using the models built by this method, they demonstrated that it was possible to distinguish seven environments having different shapes. However, the identification performance depended on the rules, which were designed before the modeling. This means that the performance of their method is unreliable if suitable rules cannot be determined.

Many of these conventional studies have created an environmental model by making raw sensor data correspond to state representations (e.g., if-then rules, landmarks, and corners). However, the performances of these methods are unreliable if suitable state representations cannot be determined.

In this paper, we present a modeling method based on state representation, which represents a change in sensory information. There have been several studies on methods of using changes in sensory information for robot state representation [7], [8], [9]. Duchon et al. used optical flow as a state representation and used it to control a mobile robot [7]. Takahashi et al. proposed a state representation based on changes in sensory output for reinforcement learning.

Nakamura et al. proposed an environmental modeling method based on changes in sensory information [9]. They used the change observed by the moving robot for the model. However, the purpose of that study was to make a model of the whole environment for use in robot action generation. The method needs a lot of data and statistical processing. Moreover, the parameters needed for statistical processing depend strongly on the environment. Such an approach is not suitable for modeling and identifying multiple environments, which is the topic treated in this paper.

In this paper, we identify an environment by using a model based on changes in sensory information. The model is built from changes in sensory data observed by the mobile robot during wall-following movements. Our method is targeted at mobile robots that have only low-sensitivity short-range sensors because there are cost advantages if such a simple robot can identify its environment. In experiments, we confirmed the effectiveness of our method by applying it to a mobile robot having only low-sensitivity infrared sensors. We found that highly accurate identification could be achieved by using our model.

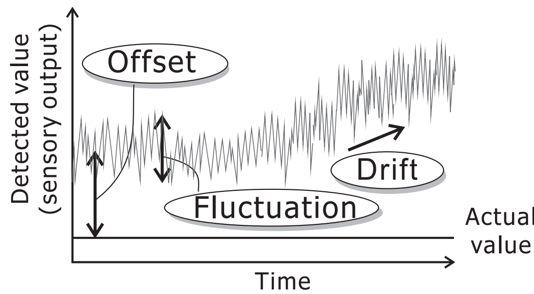

Figure 2. Sensory noise.

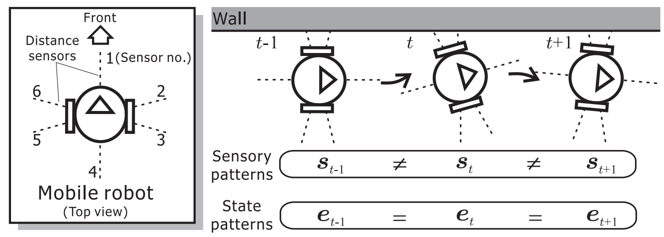

Figure 3. Dispersion of movement during wall-following.

This rest of this paper is organized as follows. Section II describes the method of modeling the environment and the method of identifying multiple environments by using the model. Section III presents experimental results obtained with a real mobile robot. Section IV discusses the experimental results. Section V concludes with a summary and mentions future work.

II. Proposed method

This section describes the method of environmental modeling and the method of identifying multiple environments by using the model. Proposing a method for even a simple robot is valuable from the engineering viewpoint. Although the method described below is for a mobile robot with only low-sensitivity infrared sensors, it should be robust against changes in lighting conditions.

A. Sensory noise

First, we discuss general sensory noise. Types of sensor noise are shown in Figure 2. Fluctuation and drift represent changes in comparatively quick and late sensory output, respectively.

Offset means the difference between an actual value and a detected value. Since sensory output generally includes such noise, frequent calibration is necessary. From the viewpoint of environment modeling, these noises reduce the similarity of models built for the same environment, which makes highly accurate identification difficult. Moreover, variations in the robot's movement also makes highly accurate identification difficult. Even if the same motor command is output from a controller, an actual robot's action will differ because of friction or mechanism factors. Therefore, even when the robot moves in response to the same-shaped wall, the obtained sensory patterns will differ (figure 3). Below, we propose an environmental modeling method that is robust against these sensory noises and the effects of movement variations.

B. State representation

We assume that the robot has multiple distance sensors, and we define the pattern $\boldsymbol{s}_t = (s_{t,1}, \cdots, s_{t,i}, \cdots, s_{t,I})$ as the sensory pattern observed at time t, where $s_{t,i}$ represents the output of sensor number i at time t. In this paper, for simplicity, we assume that all of the sensors have the same specifications. The sensor pattern series observed by the robot as it moves round the environment along a wall are used for modeling. Such a movement in which the robot goes round the environment along a wall is called a trial. And the observed sequence is defined as $\boldsymbol{s}_0, \cdots, \boldsymbol{s}_t, \cdots, \boldsymbol{s}_T$. In our method, the change in sensory patterns between times t and $t+1$ is calculated in order to remove the effect of sensory offset. The sensory pattern change is defined as $\Delta s_{t,i} = s_{t,i} - s_{t-1,i}$. Here, $\Delta s_{t,i}$ is not affected by sensory offset because it is a relative amount. Furthermore, if the rate of change of drift is very slow, $\Delta s_{t,i}$ will not be influenced by drift, either.

Next, we consider the effect of variations in the robot's behavior. Motion variation produces different values of $\Delta s_{t,i}$ even if the robot moves along the same-shaped wall. $\Delta s_{t,i}$ consists of two components of change.

$$\Delta s_{t,i} = \Delta s^f_{t,i} + \Delta s^m_{t,i}, \tag{1}$$

where $\Delta s^f_{t,i}$ is the change resulting from sensory fluctuation and $\Delta s^m_{t,i}$ is the change resulting from robot movement. $\Delta s^f_{t,i}$ is non-zero even if the robot does not move. $\Delta s^m_{t,i}$ arises from the change in actual distance between the robot and the wall caused by the robot's movement. For example, in figure 3, $\Delta s_{t,i}$ for sensors 1 to 4 consists of only $\Delta s^f_{t,i}$. Because the wall-to-robot distance in the sensing area changes with the movement, $\Delta s_{t,i}$ for sensors 5 and 6 consists of $\Delta s^f_{t,i}$ and $\Delta s^m_{t,i}$.

We assume that $|\Delta s^f_{t,i}|$ is very small compared with $|\Delta s^m_{t,i}|$. In the case of figure 3, this assumption means that $\Delta s_{t,i}$ for sensors 5 and 6 is always larger than $\Delta s_{t,i}$ for the other sensors. In our method, such a relationship between the $|\Delta s_{t,i}|$ values of sensors is used as a pattern that represents the shape of walls. A pattern $\boldsymbol{e}_t = (e_{t,1}, \cdots, e_{t,i}, \cdots, e_{t,I})$ is defined as

$$e_{t,i} = \begin{cases} 1 & \text{if } |\Delta s_{t,i}| > \eta \\ 0 & \text{otherwise} \end{cases}$$
$$(t = 1, 2, \cdots, T_e, \quad i = 1, 2, \cdots, I), \tag{2}$$

where η is a threshold and T_e is larger than T. Even if the sensory pattern varies, the same pattern $\boldsymbol{e}_t$ will be obtained

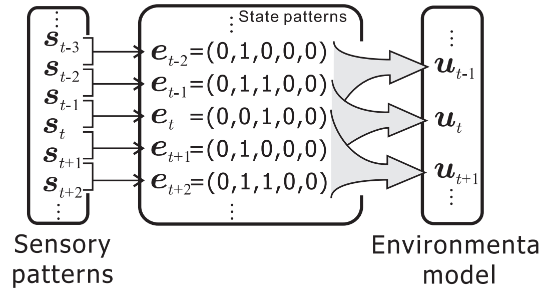

Figure 4. Proposed method ($N = 1, I = 5$).

if we set η using $|\Delta s^m_i| > \eta > |\Delta s^f_i|$ (figure 3). In this paper, we call $\boldsymbol{e}_t$ the state pattern at time t.

In Equation (2), if t is larger than T, we set $\boldsymbol{e}_t$ to a zero vector. If T_e is set large enough, state pattern sequences of the same length are acquired from environments of different sizes. The proposed state pattern uses information about the existence of a wall within sensor range. It uses the change in sensory output to detect the wall. There is also a way to detect the wall from sensory output directly. The wall can be detected if the robot judges that a wall exists when the sensory output is larger than a threshold. However, if the sensor offset differs for every trial, it is difficult to identify the existence of a wall accurately. This method needs to tune the threshold frequently. On the other hand, our method can be applied to data that includes a different offset for every trial. Additionally, the use of a different threshold η for each sensor makes our method applicable to robots with sensors having different specifications.

C. Environmental modeling and identification

The environmental modeling method is described below. First, the sequence of the state pattern is made from the sensory pattern sequence. Then, a vector $\boldsymbol{u}_k$ is calculated as

$$\boldsymbol{u}_k = \frac{1}{2N+1} \sum_{j=k}^{k+2N} \boldsymbol{e}_j$$
$$(k = 1, 2, \cdots, T_e - 2N), \tag{3}$$

where $\boldsymbol{u}_k$ is the average of $\boldsymbol{e}_t$. We defined the sequence $\boldsymbol{u}_1, \cdots, \boldsymbol{u}_k, \cdots, \boldsymbol{u}_{T_e-2N}$ as the model of the environment. The procedure for the modeling ($N = 1, I = 5$) is shown in Figure 4.

Environment identification is achieved by model comparison. Our method identifies the environment by calculating the error between the model of the current environment (test model) and the reference models.

$$E_l = \sum_{k=1}^{T_e-2N} (\boldsymbol{u}^l_k - \boldsymbol{u}^{test}_k)^2 \quad (l = 1, 2, \cdots, L), \tag{4}$$

where $\boldsymbol{u}^l_k$ and $\boldsymbol{u}^{test}_k$ are elements in the sequence of test and reference models, respectively, and E_l is the error between

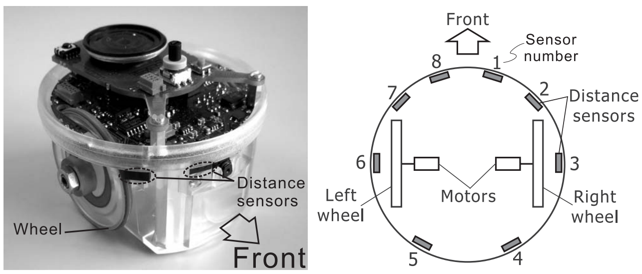

Figure 5. Mobile robot (e-puck).

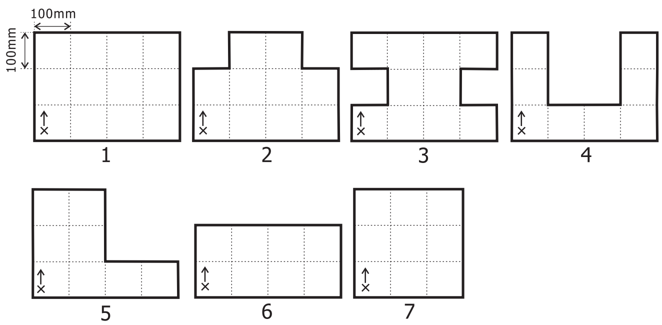

Figure 6. Environments (1–7).

the test model and the reference model of environment l. The recognition result $\hat{l}$ is given as $\hat{l} = \text{argmin}_l E_l$.

III. Experimental results and discussion

To check the effectiveness of our method, we performed experiments using a real mobile robot.

A. Experimental setup

The mobile robot used in the experiments is shown in Figure 5 [10]. It is called e-puck. It is cylindrical with a diameter of 70 mm and a height of 40 mm and it has eight distance sensors (infrared sensors) and two wheels. The sensor detection ranges are 40 mm. All the sensor readings were normalized to the range [0,1], where 1 indicates that the robot was very close to an object and 0 indicates that there was no object within the sensor's detection range. The sensor pattern $\boldsymbol{s}_t$ was denoted by a vector of order 8 ($I = 8$). Its eight elements correspond to the output values of sensors 1 to 8, respectively. The first experiment used the seven simple environments shown in Figure 6. These environments, made using polystyrene boards, consisted of straight walls and right-angled corners (figure 7). We compared the performance of our method (proposed) with AEM, which is one of the environmental identification methods proposed by Yamada and Murota [6].

AEM uses a mobile robot controlled by if-then rules, so identification is achieved using the model it makes from the sequence of rules used. For environments similar to our seven in Figure 6, they reported that their method could

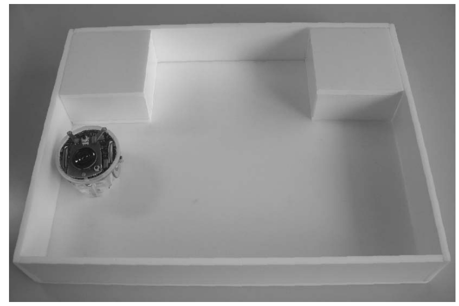

Figure 7. Experimental environment.

identify all of them accurately. In our experiments, we used a mobile robot controlled by if-then rules. The sequence of sensor patterns was obtained when the robot moved round inside the environments. The if-then rules used are shown below.

Rule A (turning in a concave corner)
If $s_8 > 0.3$.
Then turn 16 degrees clockwise.

Rule B (turning in a convex corner)
If $s_6 + s_7 < 0.2$.
Then turn 16 degrees counterclockwise and go straight for 10 mm.

Rule C (following a wall)
If $(s_6 + s_7)/2 > 0.3$.
Then turn 2 degrees clockwise and go straight for 3.3 mm.

Rule D (following a wall)
If$(s_6 + s_7)/2 \leq 0.3$.
Then turning 2 degrees counterclockwise and go straight for 3.3 mm.

In each rule, the "if" part represents the condition described by sensory output, and the "then" part represents the movement generated by the robot. At time t, the robot selects a rule from sensory pattern $\boldsymbol{s}_t$, and a movement is generated according to this rule.

The robot observed the sensory patterns and generated a behavior at every time step (0.3 s). Even when it used the same rule, its action varied as a result of friction. The robot moved according to the selected rule for 0.3 s, except when the selected rule was B, in which case the robot rotated and went straight for 0.3 s. Rules A and B were used mainly at corners, and rules C and D were used mainly when the robot moved along in accordance with straight walls.

In the experiments, we made the robot go around the environments at first and we obtained six sequences (trials) from each environment. 42 models were made for each sequence

Table I
RECOGNITION RATES FOR SEVEN ENVIRONMENTS.

Method	Maximum recognition rate
Proposed	**100%**
AEM	100%
Wall	97.6%
Sensor	93.3%

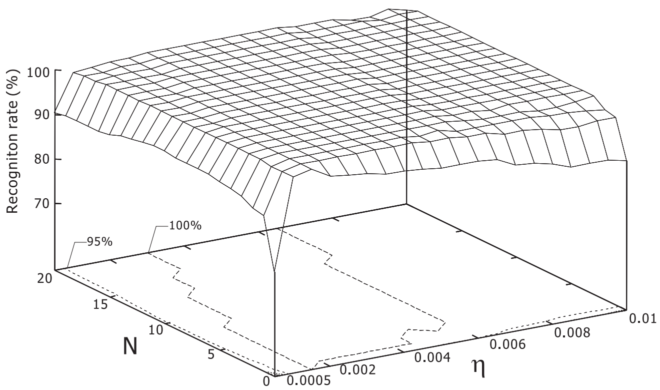

Figure 8. Recognition rates for proposed method using several parameter sets (7 environments).

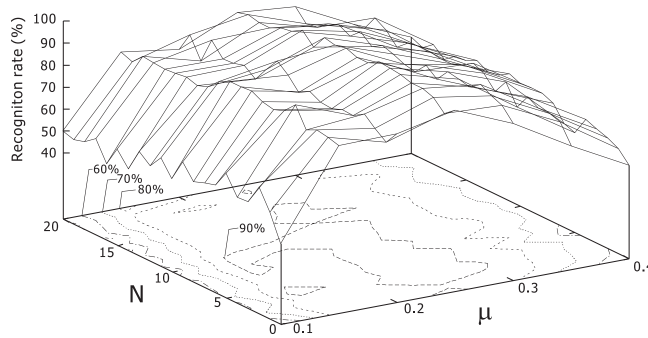

Figure 9. Recognition rates for Wall using several parameter sets (7 environments).

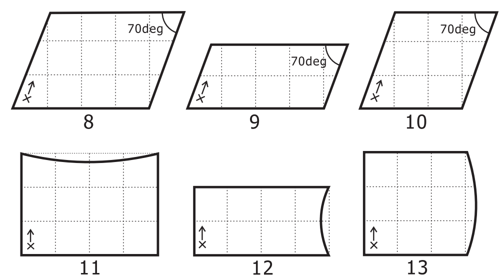

Figure 10. Environments (8–13).

(7 environments×6 trials). One model was randomly chosen for every environment and treated as a reference model. The other models were used as test models. The recognition rate was defined as the ratio of the number of tested models that were recognized correctly to the total number of tested models as follows.

$$\text{Recognition rate (\%)} = \frac{\text{Number of correctly recognized tested models}}{\text{Total number of tested models}} \times 100 \quad (5)$$

The recognition rate showed hereinafter is the average of all the rates calculated by replacing the reference model.

We assumed that there were different sensor offsets for every trial, so we added noise to the sensory patterns of each sequence. The noise level was set between 0 and 0.1 on the basis of the actual sensor offset. The maximum sequence length was 400, so we used $T_e = 450$. We also investigated the recognition rate for two other state representations: $\boldsymbol{s}_t^{WALL}$ and $\boldsymbol{s}^{SENS}$.

$\boldsymbol{s}_t^{WALL}$ was obtained by using Equation (2) with $|\Delta s_{t,i}|$ and η replaced by $s_{t,i}$ and μ, respectively. Here, $\boldsymbol{s}_t^{WALL}$ represents the existence of a wall determined using threshold μ. The sensory pattern $\boldsymbol{s}_t$ was used as $\boldsymbol{s}^{SENS}$. The methods using $\boldsymbol{s}_t^{WALL}$ and $\boldsymbol{s}_t^{SENS}$ are denoted "Wall" and "Sensor".

B. Identification of 7 environments

In this experiment, we changed parameters N, η, and μ and investigated the recognition rate. The maximum rates obtained are listed in Table I. Our method and AEM achieved 100%. The recognition rates for our method and for Wall, which were acquired by changing the parameters, are shown in Figures 8 and 9. They indicate that the recognition rate of our method does not depend on the parameters as much. The maximum rate for Wall was obtained with $\mu = 0.25$ and $N = 7$, and the maximum for Sensor was obtained with $N = 9$.

C. Identification of 13 environments

In this experiment, we investigated the recognition rate for 13 environments. We added the six new environments shown in Figure 10 to the seven environments used before. New environments 8, 9, and 10 were similar to environments 1, 6, and 8; they differed only in the corner angle. Environments 11, 12, and 13 were also similar to environments 1, 6, and 8, but each one had a curved wall. The wall curvature in environments 11, 12, and 13 were 1.2, 4.2, and 2.9 [1/m], respectively.

The maximum rates are listed in Table II. The maximum rate for our method was 99%, obtained with $\eta = 0.002 - 0.0025$ and $N = 11 - 13$. Thus, our method was able to identify the new environments. On the other hand, the maximum rate for Wall ($\mu = 0.25$ and $N = 13$) and Sensor ($N = 13$) decreased, which shows that these methods could not identify the newly added similar environments by analyzing the error in identification. The recognition rate of AEM also decreased. Almost all the mistakes occurred in

Table II
RECOGNITION RATES FOR 13 ENVIRONMENTS.

Method	Maximum recognition rate	Recognition rate using same parameters for 7 environments
Proposed	**99.0%**	**95.6%**
AEM	77.9%	-
Wall	82.3%	76.7%
Sensor	84.6%	83.3%

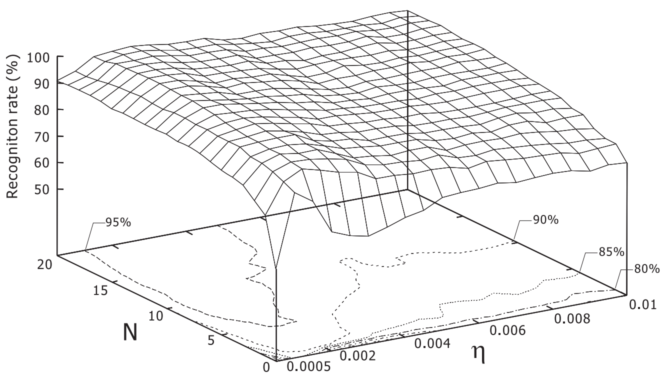

Figure 11. Recognition rates for proposed method using several parameter sets (13 environments).

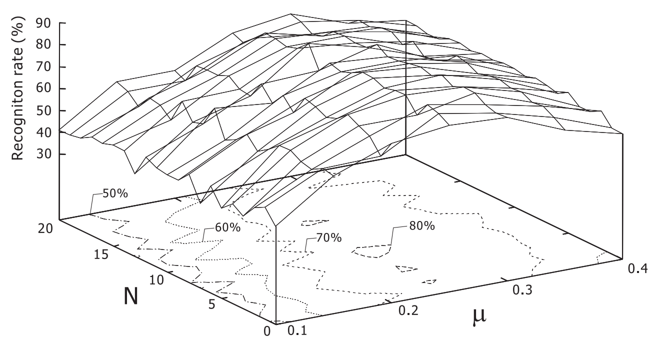

Figure 12. Recognition rates for Wall using several parameter sets (13 environments).

similar environments. When rules C and D were used, AEM considered the straight wall and modeled it. Therefore, AEM could not distinguish between the straight and curved walls. Moreover, it had trouble identifying the difference in corner angle. Although AEM can identify the corner angle from the number of times the rule is used, corner angle recognition using these rules was difficult. We conclude that deciding the rules in advance was the cause of this decline in recognition rate.

We investigated the recognition rate using the same parameters as used for the maximum rate in the seven-environment identification (Table II). Our method could distinguish them better than the other methods. This means that our method's discernment performance is robust against changes in environment. The recognition rates for our method and for Wall, which were acquired by changing the parameters, are shown in Figures 11 and 12.

Parameter η of the maximum rate decreased, as shown in Figure 11. Our method can express a state finely when the threshold η is small. However, a small η causes a decrease in recognition rate as a result of the influence of fluctuations in sensory output. It is effective to measure $\boldsymbol{s}_t$ in advance and use the results as a criterion for determining the threshold η. We infer that an appropriate threshold can be determined by increasing the parameter gradually from a small value. The determination of an appropriate value for η will be studied in future work.

In the above experiments, a designer judged how the robot went around the environments. It is possible for the robot itself to make this judgment itself if a mark is temporarily installed. Now we consider the environmental size and discernment performance. In AEM, the weights are set up for every rule and a model is made using the value accumulated in the order of the rule series. This means that the accumulation-based AEM tends to be influenced by the environment becoming large. On the other hand, our method is not easily affected. Therefore, our method will be more effective than AEM when the environment becomes large.

D. Discernment with a random start position

We confirmed the effectiveness of our method when the start position was set at random. In this experiment, we set the start position randomly in each trial and obtained the sensory pattern sequence. The matching error was calculated by shifting the sensor patterns by one step at a time and creating a test model; the identification result was then acquired from the minimum error. The experiment was performed for both 7 and 13 environments. Threshold $\eta = 0.003$ and $N = 10$ were used in the 7-environment identification and $\eta = 0.002$ and $N = 14$ were used in the 13-environment identification. The recognition rates for 7- and 13-environment identification were 97.6% and 95.1%, respectively. These results show that our method is also effective when the start position is random.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a modeling method based on state representation, which represents a change in sensory information. This model enables a mobile robot to identify its environment. Experiments on a real mobile robot having only low-sensitivity infrared sensors showed the effectiveness of our method. A comparison between our method and a conventional one showed that ours had higher performance.

In our method, the size of the robot and the arrangement of the sensors and the sensing area affect the model. We will investigate these effects in the future. Our method assumes that the environment is a closed area. Therefore, it cannot identify an environment where, for example, the landmark is in the center of the room. For such a case, a motion that is not along a wall is needed. Moreover, it cannot handle an object that moves into the environment: for this case, we must propose a state representation based on information about the robot's motion. We plan to investigate these issues in the future.

ACKNOWLEDGMENT

This research has been supported by funding from the Kayamori Foundation of Information Science Advancement, the Artificial Intelligence Research Promotion Foundation, the Nakajima Foundation and KAKENHI (No. 21700219).

REFERENCES

[1] M. J. Matarić, "Integration of representation into goal-driven behavior-based robots," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 304–312, 1992.

[2] Y. Matsumoto, M. Inaba, and H. Inoue, "View-based navigation using an omniview sequence in a corridor environment," *Machine Vision and Applications*, vol. 14, p. 121–128, 2003.

[3] H. Iwasa, N. Aihara, N. Yokoya, and H. Takemura, "Memory-based self-localization using omnidirectional images," *Systems and Computers in Japan*, vol. 34, no. 5, pp. 56–68, 2003.

[4] K. Nishida, T. Tanaka, and T. Kurita, "Constructing a map of place cells for mobile robot navigation," in *International Congress Series 1269*, pp. 189–192, 2004.

[5] U. Nehmzow and T. Smithers, "Mapbuilding using self-organizing networks in really useful robots," in *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pp. 152–159, 1991.

[6] S. Yamada and M. Murota, "Unsupervised learning to recognize environments from behavior sequences in a mobile robot," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA98)*, pp. 1871–1876, 1998.

[7] A. P. Duchon, L. P. Kaelbling, and W. H. Warren, "Ecological robotics," *Adaptive Behavior*, vol. 6, no. 3-4, pp. 473 – 507, 1998.

[8] Y. Takahashi, M. Asada, and K. Hosoda, "Reasonable performance in less learning time by real roobot based on incremental state space segmentation," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS'96)*, pp. 1518–1524, 1996.

[9] T. Nakamura, S. Takamura, and M. Asada, "Behavior-based map representation for a sonor-based mobile robot by statistical methods," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'96)*, pp. 276–283, 1996.

[10] http://www.e-puck.org/.