The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Observation Planning for Environment Information Summarization with Deadlines

Hiroaki Masuzawa and Jun Miura
Department of Computer Science and Engineering
Toyohashi University of Technology

Abstract— **Mapping is an activity of making a useful description of an environment. Not only geometric information such as free space shape but also semantic information such as object names are sometimes important. We call such a map making *environment information summarization* because how to summarize may change depending on the purpose of the map and the context. One important aspect of such summarization is the deadline, which imposes a time constraint on the robot's mapping activity. We therefore develop an observation planning method for the summarization with deadlines. The method can cope with various types of deadlines specified in the form of loss function. Experimental results shows various robot behaviors are generated by only changing the deadline specification.**

Index Terms— **Environment information summarization, observation planning, planning with deadline, mobile robots, object appearance model.**

I. Introduction

Environment modeling is one of the active research areas in robotics. One important aspect of environment modeling is *geometric mapping* [1], which generates free space maps or landmark maps. Many successful mapping or SLAM methods have been developed which use statistical tools such as Kalman filter [2], [3] or particle filter [4], [5].

A recent trend in mapping is *semantic mapping*, which generates maps describing not only geometric properties but also semantic information such as object recognition results and space segmentation and categorization [6], [7], [8], [9], [10]. Such a semantic map is suitable as a medium for human-robot communication; the user can easily give location information to the robot using such semantic information.

We have been using the term *environment information summarization* to represent the activity of making such a map (or a *summary*) because what kinds of information are included in the summary depends on the purpose of the map and the context (e.g., deadline). We have developed an environment information summarization system that can generate a map describing free spaces and placements of specified objects. The system features an observation planning method for efficient summarization [11].

This paper deals with an environment information summarization with *deadlines*. It is often necessary to consider deadlines in automatic mapping by mobile robots; only a limited time may be granted to a mobile robot for surveying an area or such a robot may has a limited working time with batteries.

Observation planning problems appear in various contexts of mobile robotics such as exploration planning [12], [13] and object search strategy planning [14], [15]. Some combines these two types of planning [10]. Our previous planner also dealt with both mapping and object search [11]. It alternately repeats two kinds of viewpoint selection, one is for observing unknown regions and the other is for verifying object candidates found so far. The former planning is performed by a heuristic greedy algorithm, while the latter is formulated as an optimization problem of minimizing the expected cost for the verification. Since the two planning steps are independent of each other, however, it is difficult to optimize the whole observation activity and/or to consider deadlines.

In this paper, therefore, we develop a unified planner for an environment information summarization. To this end, we hierarchically decompose the observation planning problem into two levels; the higher level determines the optimal order of observation of *groups*, which is either an object candidate (or a set of object candidates) or an unknown region, and the lower level actually determines the optimal viewpoint sequence for a group. Based on this decomposition, accompanied with a loss function representation of deadlines, our planner can generate an efficient observation plan on-line considering the deadline.

The rest of the paper is organized as follows. Section II describes the problem treated in this paper and an overview of the proposed method. Section III briefly explains the object appearance models and object recognition. Section IV describes the planning method in detail. Section V shows experimental results using a humanoid robot. Section VI concludes the paper and discusses future work.

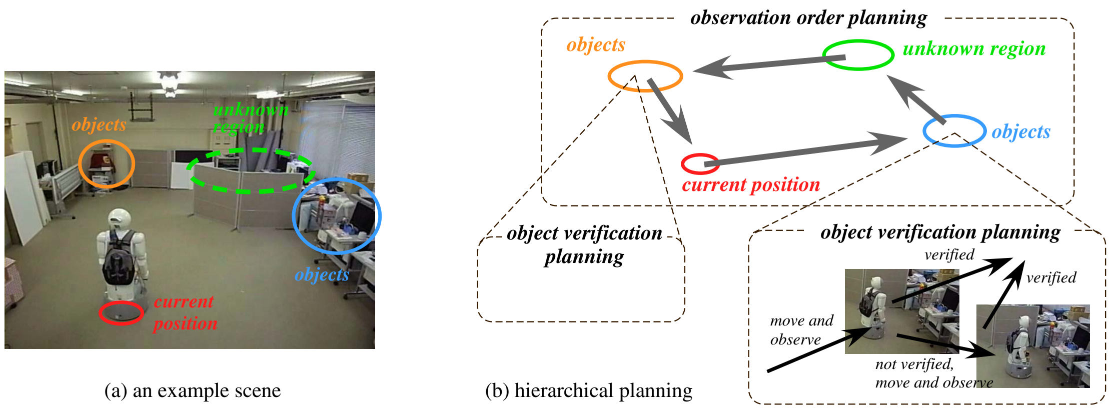
II. Overview of the observation planning

A. Environment information summarization problem

The environment information summarization treated in this paper is as follows [11]. The robot is required to (1) make a free space map and (2) detect specified objects and record their positions in the map. We use a humanoid robot that has a laser range finder (LRF) to detect free spaces and vision to detect objects.

As prior knowledge, the robot is given a set of appearance models of objects to recognize and the shape of the room to summarize. The deadline for summarization, which is an

978-1-4244-6676-4/10/$25.00 ©2010 IEEE



(a) an example scene (b) hierarchical planning

Fig. 1. Hierarchical decomposition of the planning problem.

important difference between the previous and the current work, is also given to the robot.

There are two types of observation actions. One is to observe an unknown region for updating a free space map using the LRF and for detecting object candidates using color information. The other is to verify object candidates using a local feature-based matching.

B. Two-level observation planning

Every time new information is obtained by observation, the robot updates the *state*, from which it knows the size and the position of unknown regions and the locations of object candidates; these are the target of observation and an appropriate observation action is selected for each target. Either of an unknown region or an object candidate (or a group of neighboring object candidates) constitutes a group.

The whole observation planning is hierarchically decomposed into two levels (see Fig. 1): *observation order planning* and *object verification planning*. In the observation order planning, the robot determines the optimal observation order of the groups, based on the value of observing each group and the cost of observation, with consideration of the deadline. In the case where a group is composed of an object or a set of objects, the object verification planning is performed for calculating the value and the cost of verification, with generating an optimal sequence of viewpoints.

The robot alternately repeats the planning and the execution step. Although the robot makes a full plan at each planning step, only the observation of the first group is performed because new information may will be obtained which alters the remaining part of the plan.

At the initial position, which is usually at the entrance of a room to summarize, the robot is set to observe several directions in order to find all object candidates visible from that position and make an initial free space map, thereby determining the *initial state*.

III. Object Appearance Model for Recognition

This section briefly explains object recognition using object appearance models and the calculation of *predicted recognition probability* used in observation planning. Refer to [11] for more details.

Fig. 2. Object recognition example: candidates (red) and recognized (green).

object A object B object C

Fig. 3. Three objects used.

A. Appearance model

Object recognition is done in two stages: object candidate detection using color histogram and object verification using SIFT feature [16]. Figure 2 shows an example of candidate detection and verification for object A shown in Fig. 3.

SIFT is robust to scale changes to some extent. If the distance from the robot to an object becomes large, however, an enough number of SIFT matches may not be obtained. To consider the trade-off between the cost of observation (i.e., the travel distance to approach an object) and the quality of recognition, we examine this distance-dependency of the SIFT-based matching. We also examine the effect of the relative angle between an object surface and the camera optical axis. Based on these examinations, we develop an appearance model which is to predict the number of SIFT matches for a given observation condition. An object has in general different appearances for different surfaces, for each of which an appearance model is constructed.

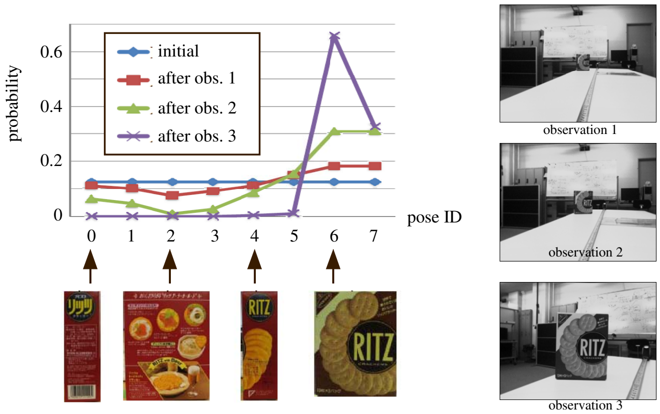


Fig. 4. Pose estimation example.

B. Object orientation estimation and predicted recognition probability

The effective distance for recognizing an object depends on what surface of the object is visible. So we on-line estimate the object orientation using the appearance models. Figure 4 shows an pose estimation experiment. We discretize the orientation to eight. By observing the object while approaching it, the probability distribution gradually converges.

The number of SIFT matches could be calculated if we knew the object orientation, but in reality, we can only know its distribution. We thus predict the probability of successful recognition $P_{recog}(\boldsymbol{X}_c, \boldsymbol{X}_{obj})$ from the estimated distribution. This probability is defined for viewpoint $\boldsymbol{X}_c$ and object position $\boldsymbol{X}_{obj}$ and calculated as the one that the number of SIFT matches exceeds a threshold. This value is used for the verification planning described below.

IV. Observation planning with deadline

The observation planning treated in this paper has two levels: the *observation order planning* and the *object verification planning* (see Fig. 1). This decomposition is realized by generating *observation groups*. This section describes the group generation and the two kinds of planning.

A. Observation group generation

There are two kinds of the target of observation in the current environment information summarization: unknown regions and object candidates. The robot observes an unknown region to update the free space map and to detect object candidates, while it observes object candidates for verifying them. We thus discriminate them and generate observation groups as follows.

- An unknown region constitutes a group.
- An isolated object candidate or a set of neighboring object candidates constitutes a group. If multiple candidates can be recognized from a viewpoint, by considering the maximum recognizable ranges of the candidates, then they are considered neighboring.

Examples of observation groups are shown in Fig. 7.

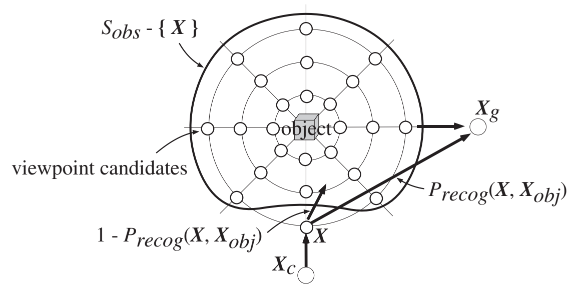


Fig. 5. One object verification.

B. Object verification planning

The algorithm for object verification planning basically follows the one developed in our previous paper [11]. This subsection explains the outline of the method.

Figure 5 shows the case where the robot at position $\boldsymbol{X}_c$ makes a plan for verifying one object candidate and moving to position $\boldsymbol{X}_g$ after verification. Let $\boldsymbol{X}$ be the position for verification. We define the set S_{obs} of possible viewpoints for verification as the 24 points on three co-centric circles, the largest of which has the radius equal to the maximum recognizable range.

The criterion used for object verification planning is to minimize the expected time for observation and movement. Let $t^*(\boldsymbol{X}_s, \boldsymbol{X}_g, obj, S_{obs})$ be the minimum expected time for the verification action of object obj with the remaining viewpoint set S_{obs} in the case where the robot is now at $\boldsymbol{X}_s$ and the next subgoal is $\boldsymbol{X}_g$. This minimum expected time is defined by the following recurrence formula:

$$
\begin{aligned}
&t^*(\boldsymbol{X}_s, \boldsymbol{X}_g, obj, S_{obs}) \\
&= \min_{\boldsymbol{X} \in S_{obs}} \left[\begin{array}{c} \dfrac{dist(\boldsymbol{X}_s, \boldsymbol{X})}{v_{robot}} + t_{verify} \\ +P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj}) \dfrac{dist(\boldsymbol{X}, \boldsymbol{X}_g)}{v_{robot}} \\ +(1 - P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj})) \cdot \\ t^*(\boldsymbol{X}, \boldsymbol{X}_g, S_{obs} - \{\boldsymbol{X}\}) \end{array} \right], \quad (1)
\end{aligned}
$$

where $P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj})$ is the predicted recognition probability (see Sec. III-B), t_{verify} is the time for one verification action, v_{robot} is the average speed of the robot.

A naive calculation of this equation is costly because t^*'s are calculated for many times with similar $\boldsymbol{X}$'s. We therefore make look-up tables for several patterns of viewpoint set. Each table is referred to by four parameters: start position, goal position, viewpoint pattern, and the object orientation.

For the case of a group with multiple object candidates, we consider two kinds of observation strategies. The sequential strategy makes observations of the candidates sequentially, while the parallel strategy uses viewpoints from where the candidates can be observable. We compare the two strategies and select the better one.

C. Observation order planning

The observation order planning determines the order of observation considering the values of observing objects and unknown regions and the cost imposed by the deadline.

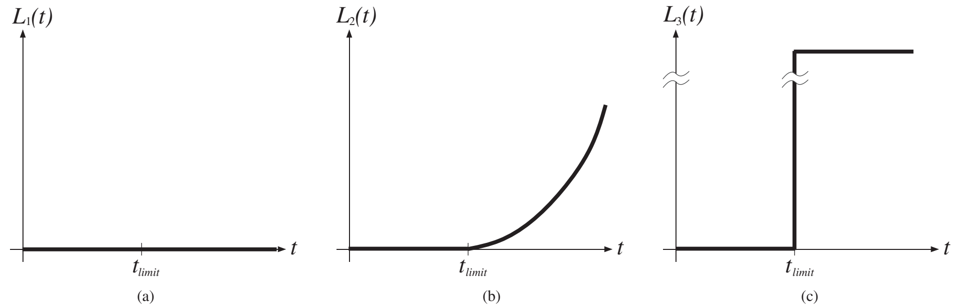


Fig. 6. Loss functions representing deadlines: (a) no deadline. (b) soft deadline. (c) hard deadline.

1) Utility for observation order: Suppose there are N observation groups and $\mathcal{G} = \{G_1, G_2, \ldots, G_N\}$ be an order of observations. The utility $U(\mathcal{G})$ of $\mathcal{G}$ is defined by

$$U(\mathcal{G}) = V(\mathcal{G}) - L\left(t_{mov}(\mathcal{G}) + \sum_{i=1}^{N} t_{obs}(G_i)\right), \quad (2)$$

where V is the value of observing groups, L is a loss function imposed by violating the time limit, t_{mov} is the time for moving between groups, and t_{obs} is the time for observing a group.

2) Time for movement between groups: Each group has a set of entry points. For an object (or an object set), entry points are the viewpoint candidates on the outermost circle for an object. For an unknown region, they are selected with a regular interval on the boundary between the unknown and the known region.

To calculate the time of moving from one group to another, we first calculate the minimum-length path for every pair of entry points from the two groups by a robot path planner, and then select the shortest one. The time for movement is given by dividing the minimum path length by the average robot speed.

3) Time for observing a group: When a group is composed of object candidates, the time for observing it is calculated by the object verification planning (see eq. (1)).

When a group is an unknown region R, the time for observing it is estimated as the sum of those for free space map updating, t_{map}, and for verifying object candidates to be detected in the region, $\hat{t}_{obj}(A)$. The former is set to a constant value because the size of the region has little effect on the map updating process. The latter is given by

$$\begin{aligned}\hat{t}_{obj}(A) \;&=\; \frac{|R|}{A_U}\left\{\sum_{j=1}^{M} \bar{n}_j t^*(\boldsymbol{X}_s, \boldsymbol{X}_g, obj_j, S_{obj_j})\right. \\ &\qquad \left. + \bar{t}_{mov}(\sum_{j=1}^{M} \bar{n}_j - 1)\right\}, \quad (3)\end{aligned}$$

where t^* is the estimated time for verifying an object (see eq. (1)), $|R|$ is the area of region R, A_U is the total area of currently unknown regions, M is the number of specified objects to be modeled, $\bar{n}_j$ is the expected number of obj_j per room, $\bar{t}_{mov}$ is the averaged time for moving between objects for verification.

4) Deadlines: Environment information summarization tasks usually have deadlines. For example, the robot used for summarization have a limited amount of battery or may be granted a limited time for summarization before deployment; a typical task could be "make a summary of this room within 30 minutes." Such various deadlines are roughly classified into two categories. One is *hard* deadline; the loss of violating it is very large or sometimes infinity. The other is *soft* deadline; the loss gradually increases as the time passes beyond the deadline. Miura et al. [17] presented several types of soft deadlines for the arrival time used by an intelligent car navigation system. We take a similar approach to represent soft constraints.

In this paper, we consider the following three types of loss functions representing deadlines (see Fig. 6):

$$L_1(t) \;=\; 0, \quad (4)$$

$$L_2(t) \;=\; \begin{cases} 0 & (t < t_{limit}) \\ k_{limit}(t - t_{limit})^2 & (\text{otherwise}) \end{cases}, \quad (5)$$

$$L_3(t) \;=\; \begin{cases} 0 & (t < t_{limit}) \\ \infty & (\text{otherwise}) \end{cases}, \quad (6)$$

where t_{limit} is the deadline and k_{limit} is a constant.

5) Value of observing a group: The value of observing a group is given by

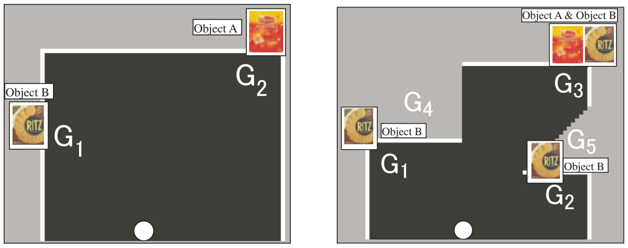
$$V(\mathcal{G}) = \sum_{i=1}^{N} V_{obs}(G_i), \quad (7)$$

$$V_{obs}(G_i) = \begin{cases} \sum_{Obj \in G_i} V_{obj}(Obj) \\ \quad (G_i \text{ is a group of object candidates.}) \\ V_{unk}(G_i) \\ \quad (G_i \text{ is an unknown region.}) \end{cases} \quad (8)$$

$$V_{unk}(G_i) = k_{area}|R_{G_i}| + \frac{|R_{G_i}|}{A_u}\sum_{j=1}^{M} \bar{n}_j V_{obj}(obj_j), \quad (9)$$

where $V_{obj}(Obj)$ is the value of object Obj to be obtained by verifying it and $|R_{G_i}|$ is the area of the unknown regions in G_i. The first term of V_{unk} is the value of observing an unknown region and is estimated to be proportional to the size of the unknown region. The second term is the expected value of verifying specified objects to be found there.

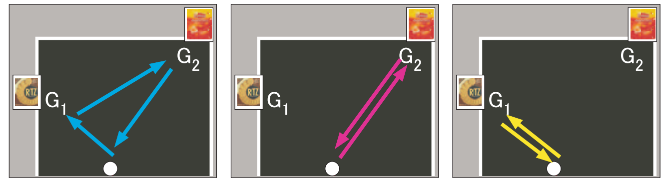
6) Calculating the best order: The best order is currently found by an exhaustive search because the number of groups



(a) environment 1

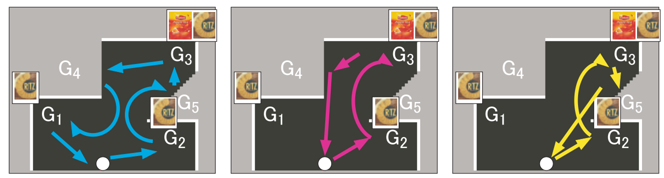
(b) environment 2

Fig. 7. Simulation settings.



(a) $L_1(t)$ (b) $L_2(t)$ (c) $L_3(t)$

Fig. 8. Results of observation order planning for Fig. 7(a) for the three loss functions.



(a) $L_1(t)$ (b) $L_2(t)$ (c) $L_3(t)$

Fig. 9. Results of observation order planning for Fig. 7(b) for the three loss functions.

to consider at a time is not large (usually two or three) in the current experimental setting. Several approximate methods can be adopted in a more complex environment.

V. Experimental Results

We conducted experiments of summarizing a $7.0\,[m] \times 5.5\,[m]$ room. The specified objects to find are the ones shown in Fig. 3. Table I indicates the values and the expected numbers of the objects.

A. Simulation of observation order planning

Figure 7 shows two simulation settings. Figure 7(a) is a simple environment with no unknown regions and two object candidates. The white circle at the bottom indicates the robot's start and goal position. There are three distinct plans depending on what object groups to observe (G_1, G_2, and $G_1\&G_2$). Their values, times needed, losses, and utilities for the three loss functions are summarized in Table II. Figure 8 compares three *different* plans generated for the three loss functions for the *same* problem. For $L_1(t)$, a plan for observing both objects is generated. For $L_2(t)$, a higher utility is obtained by observing a more valuable Object A than Object B at the cost of travelling a longer distance. For $L_3(t)$, a plan for observing the nearer object is generated in order to respect the hard deadline.

Figure 7(b) is a more complex environment with four object candidates grouped into three (G_1, G_2, and G_3) and two unknown regions (G_4 and G_5). Figure 9 shows the results of observation order planning for respective loss functions. The looser the deadline is, the more objects are chosen in the observation plan. The total time for planning for this setting is about $900\,[ms]$; this is reasonably fast for on-line planning.

TABLE I

The values and the expected numbers of objects.

	Object A	Object B	Object C
Value $V(obj)$	100	50	10
Expected Number $\bar{n}_{obj}$	1.0	2.0	1.0

TABLE II

Simulation results for Fig. 7(a).

Plan	Value	Loss L			Utility U		
$\mathcal{G}$	V	L_1	L_2	L_3	L_1	L_2	L_3
G_1	50	0	0	0	50	50	50
G_2	100	0	2.76	∞	100	97.24	$-\infty$
$G_1 \rightarrow G_2$	150	0	296.26	∞	150	-146.26	$-\infty$

B. Environment information summarization experiments using a real robot

We conducted environment information summarization experiments using a real robot. The robot uses a laser range finder for free space mapping and a stereo vision for object detection and verification. Figure 10 illustrates the setting for the experiments and Table III indicates the values for the three observation groups (Object B on the left, Objects A and C on the right, the unknown region behind the partition). Experimental scenes can be seen on an accompanying video.

1) Experiment without deadline: Figure 11 shows the process of summarization without deadline (i.e., using $L_1(t)$). Snapshots, planned actions, and summarization results/verified objects during the experiments are shown in the left, center, and the right column, respectively. In the grid map, black, white, and gray cells indicate free space, obstacles, and unknown regions, respectively. The robot position is represented by a red circle. The green and the blue circles indicate the object candidate positions and verified object positions, respectively. It took about $150\,[s]$ to make a summary describing a complete free space map with the description of all three objects.

TABLE III

Values for observation groups.

	Object A & Object C	Object B	Unknown region
Value	110	50	950

(a) Experimental scene. (b) Object placements.

Fig. 10. Experimental settings.

2) Experiment with soft deadline: Figure 12 shows the process of summarization with soft deadline (i.e., using $L_2(t)$). t_{limit} is set to $60\,[s]$. The robot made a plan for observing a group of two objects on the right and the unknown region, at the cost of exceeding t_{limit}. It took about $102\,[s]$ to make a summary describing a complete free space map with the description on two objects and one candidate.

3) Experiment with hard deadline: Figure 13 shows the process of summarization with hard deadline (i.e., using $L_3(t)$). t_{limit} is set to $60\,[s]$. Although the robot found all object candidates, due to the limited time, it observed only the unknown region. It took about $67\,[s]$ to make a summary describing a complete free space map and three object candidates. The total execution time exceeded a little the deadline due to an optimistic estimation of travel time.

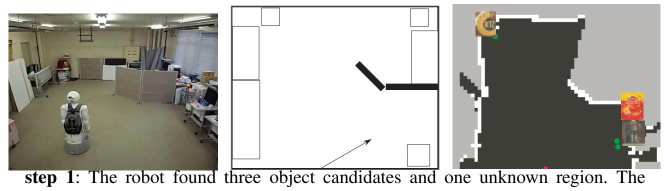
VI. Conclusions and Future Work

This paper has described an observation planning method for environment information summarization with deadlines. The summarization problem treated in this paper is to generate a free space map with placements of specified objects. The whole planning problem is hierarchically decomposed into two subproblems: an observation order planning which considers the value and the cost of observation action and deadlines and object verification planning which considers the uncertainty of objects' states. The proposed method has been successfully applied to a real summarization problem using a humanoid robot with vision and range sensors. Various robot behaviors were generated using the same planner only by changing the deadline specifications.

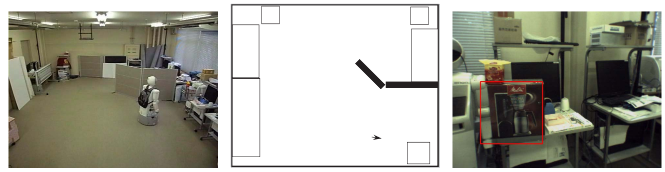
A future work is to add visual features and object models to cope with more complex environments with various objects and rooms. Developing a more efficient planner based approximate on-line algorithms is also future work. We are currently pursuing on-line approximation algorithms for observation order planning [18]. Introducing such algorithms will enable the robot to summarize large environments efficiently.

Acknowledgment

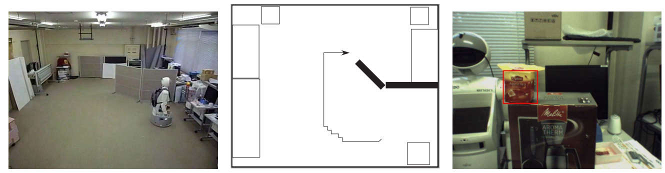
This work is supported by NEDO (New Energy and Industrial Technology Development Organization, Japan) Intelligent RT Software Project.



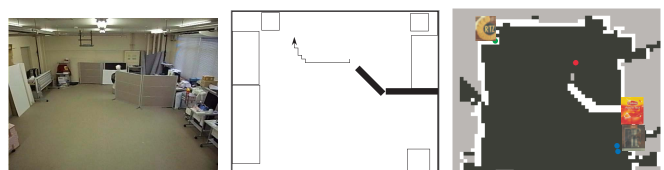
step 1: The robot found three object candidates and one unknown region. The two candidates on the left are grouped. The plan for observing all groups in a counter-clockwise way was generated and the robot moved to the first group.



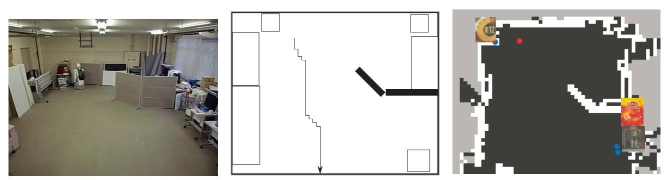
step 2: The robot observed the two objects at a viewpoint. Since one of them was verified successfully, the robot made a plan to verify the other and executed it.



step 3: The other object was verified successfully. the robot then made a plan to observe the unknown region and the other object in this order and executed it.



step 4: The robot observed the unknown region and updated the map. Since no object candidates were found in this new region, the robot made a plan to verify the remaining candidate and executed it.



step 5: The robot finished verifying all objects and made and executed a plan to go back to the initial position.

step 5: The robot finished the summarization. The summary includes the free space map and the description of all objects (blue circles).

Fig. 11. Summarization using $L_1(t)$.

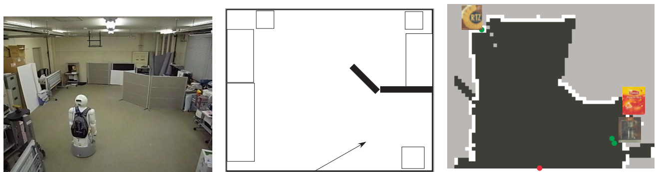
References

[1] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, 2005.

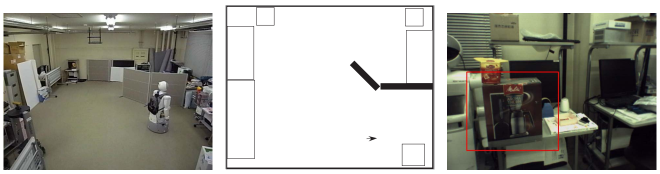
[2] J.J. Leonard and H.F. Durrant-Whyte. Simultaneous Map Building and Localization for an Autonomous Mobile Robot. In *Proceedings of 1991 IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, pp. 1442–1447, 1991.

[3] J.E. Guivant and E.M. Nebot. Optimization of the Simultaneous Localization and Map-Building Algorithm for Real-Time Implementation. *IEEE Trans. on Robotics and Automation*, Vol. 17, No. 3, pp. 242–257, 2001.

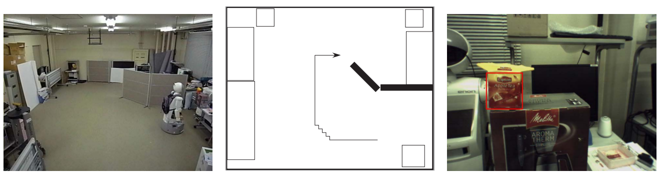
[4] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping



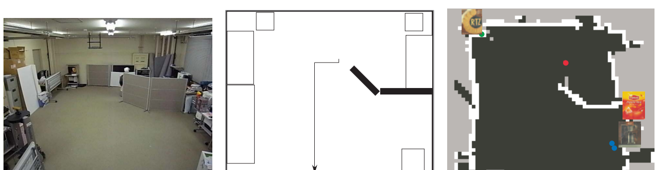
step 1: The robot found three object candidates and one unknown region. The plan for observing two objects on the right and the unknown region was generated and the robot moved to the first group.



step 2: The robot observed the two objects at a viewpoint. Since one of them was verified successfully, the robot made a plan to verify the other and executed it.



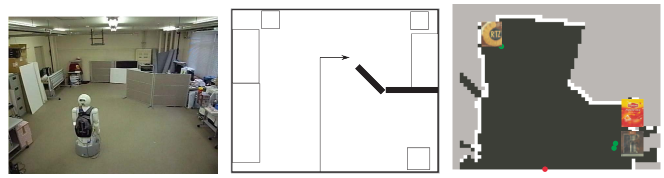
step 3: The other object was verified successfully. the robot then made a plan to observe only the unknown region and executed it.



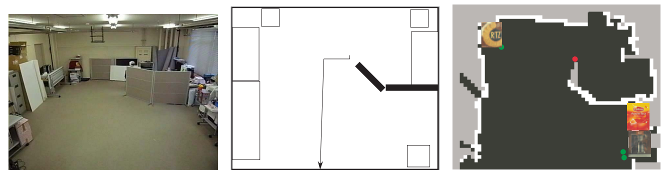
step 4: The robot observed the unknown region and updated the map and found no object candidates there. Since observing the other object led to a large loss, the robot made a plan to go back to the initial position and executed it.

step 5: The robot finished the summarization. The summary includes the free space map and the description of two objects (blue circles) and one object candidate (green circle).

Fig. 12. Summarization using $L_2(t)$.



step 1: The robot found three object candidates and one unknown region. The robot made and execute a plan for observing only the unknown region.



step 2: The robot observed the unknown region and updated the map. Since no object candidates were found in this new region, the robot made a plan to go back to the initial position and executed it.

step 3: The robot finished the summarization. The summary includes the free space map and three object candidates (green circles).

Fig. 13. Summarization using $L_3(t)$.

Problem. In *Proceedings of AAAI National Conf. on Artificial Intelligence*, pp. 593–598, 2002.

[5] G. Grisetti, C. Stachniss, and W. Burgard. Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling. In *Proceedings of 2005 IEEE Int. Conf. on Robotics and Automation*, pp. 2432–2437, 2005.

[6] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J.A. Fernández-Madrigal, and J. González. Multi-Hierarchical Semantic Maps for Mobile Robotics. In *Proceedings of the 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 3492–3497, 2005.

[7] M. Tomono. 3-D Object Map Building Using Dense Object Models with SIFT-Based Recognition Features. In *Proc. of the 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1885–1890, 2006.

[8] Z. Zivkovic, O. Booij, and B. Kröse. From Images to Rooms. *Robotics and Autonomous Systems*, Vol. 55, No. 5, pp. 411–418, 2007.

[9] S. Vasudevan, S. Gächter, V. Nguyen, and R. Siegwart. Cognitive Maps for Mobile Robots – An Object Based Approach. *Robotics and Autonomous Systems*, Vol. 55, No. 5, pp. 359–371, 2007.

[10] D. Meger, P.-E. Forssén, K. Lai, S. Helmer, S. McCann, T. Southey, M. Baumann, J.J. Little, and D.G. Lowe. Curious George: An Attentive Semantic Robot. *Robotics and Autonomous Systems*, Vol. 56, No. 6, pp. 503–511, 2008.

[11] H. Masuzawa and J. Miura. Observation Planning for Efficient Environment Information Summarization. In *Proceedings of 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 5794–5780, 2009.

[12] A.A. Makarenko, S.B. Williams, F. Bourgault, and H.F. Durrant-Whyte. An Experiment in Integrated Exploration. In *Proceedings of 2002 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 534–539, 2002.

[13] V.A. Sujan and M.A. Meggiolaro. Intelligent and Efficient Strategy for Unstructured Environment Sensing Using Mobile Robot Agents. *J. of Intelligent Robotic Systems*, Vol. 43, pp. 217–253, 2005.

[14] Y. Ye and J.K. Tsotsos. Sensor Planning for 3D Object Search. *Computer Vision and Image Understanding*, Vol. 73, No. 2, pp. 145–168, 1999.

[15] F. Saidi, O. Stasse, K. Yokoi, and F. Kanehiro. Online Object Search with a Humanoid Robot. In *Proceedings of 2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1677–1682, 2007.

[16] D.G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. of Computer Vision*, Vol. 60, No. 2, pp. 91–110, 2004.

[17] J. Miura, M. Itoh, and Y. Shirai. Towards Vision-Based Intelligent Navigator: Its Concept and Prototype. *IEEE Trans. on Intelligent Transportation Systems*, Vol. 3, No. 2, pp. 136–146, 2002.

[18] M. Boussard and J. Miura. Observation Planning with On-line Algorithms and GPU Heuristic Computation. In *Proceedigns of ICAPS-2010 Workshop on Planning and Scheduling under Uncertainty*, 2010.