Proc. 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems,
pp. 5794-5800, St. Louis, MO, USA, Oct. 2009.

# Observation Planning for Efficient Environment Information Summarization

Hiroaki Masuzawa and Jun Miura
Department of Information and Computer Sciences
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 but also object placements are important if the map is used for human-robot communication. We call such a map making *environment information summarization* because how to summarize may change depending on the purpose of the map. Environment information summarization usually includes searching for specified objects in the environment. It is, therefore, crucial to make a good observation plan for efficient summarization. We develop an observation planning method which uses object appearance models for appropriately handling a trade-off between visual data quality and vision cost. Experimental results using a vision-based humanoid robot show the effectiveness of the proposed planning method.**

***Index Terms*— Environment information summarization, observation planning, 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 mapping [1]. Many successful mapping or SLAM methods have been developed which use statistical tools such as Kalman filter (e.g., [2]) or particle filter (e.g., [3]). A major objective of these methods is to make an accurate free space map. Such a *geometric* map is important in planning a safe robot motion in the mapped environment.

When maps are used for natural human-robot communication, *semantic information* is also important. For example, we would like to direct a robot to move to some specific place or the place where some specific object exists not by using the coordinates of the destination but by using the place or object name. Semantic mapping includes object recognition by mobile robots [4] and space segmentation and categorization [5].

Several robot systems have also been developed that can extract semantic information while they move around. Vasudevan et al. [6] proposed an object-based representation and mapping of semantic information in indoor environments. Each place (room or corridor) is characterized by placement of known objects there. The robot can learn places and their relationships and recognize previously-visited places. This work does not deal with planning for efficient semantic mapping. Galindo et al. [7] used a conceptual hierarchy in which several concepts (objects and places) are organized using “is-a” and “has” semantic links. The hierarchy is used for obtaining a map which describes objects, places, and their relationships. Since the spatial relationships are also represented in the map, a plan can be generated which navigates the robot from the current place to the destination. The work does not, however, consider the efficiency of the mapping process.

If we use a map as a medium for human-robot communication, the map needs to include both geometric and semantic information. In this paper, we use the term *environment information summarization* to represent the activity of making such a description (or a *summary*) because what kinds of information are included in the summary depends on the purpose of the map. An example type of summary, treated in this paper, includes a rough spatial configuration of a room and placement of important objects in the room; a user may use the summary to direct the robot a task of fetching something, while the robot may use it for autonomous navigation. Observation planning is a key to realize an efficient summarization.

Observation planning problems appear in various contexts of mobile robotics. Makarenko et al. [8] proposed *integrated exploration* that balances three kinds of utilities, information gain, navigation cost, and localization quality, for an efficient and reliable mapping. Sujan and Meggiolaro [9] used an information gain criterion for exploring an unknown region with multiple robots. These works focus not on search for specific objects but on exploring unknown regions. Wang et al. [10] dealt with a view planning problem which considers both the observation cost for inspecting the surfaces of objects and the traveling cost between viewpoints. This work does not consider observation uncertainty.

Tsotsos’s group [11], [12] has proposed a general framework for solving a visual object search problem. Using the probabilistic distribution of the target position and the probabilistic detection functions, the object search problem has been formulated as a statistical optimization problem. Saidi et al. [13] proposed a similar approach to a 3D object search using a humanoid robot. These works deal with the problem of search for one object in the environment.

In this paper, we consider the following scenario. A robot is ordered to examine an approximate shape of the free space in a room and the positions of specified objects there. The robot is not given what objects exist and how many. It needs to enter the room, explore the room while detecting objects and planning motions, and exit. In general, it is better to approach an object closely in order to recognize the object more reliably. If the object position and the next subgoal of the robot are far apart from each other, however, two objectives, increasing the recognition probability and reducing the path length to the goal, may conflict. Miura and

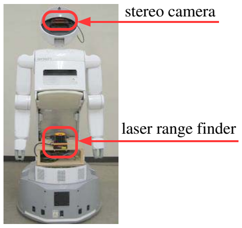



Fig. 1. The robot with stereo vision and laser range finder.

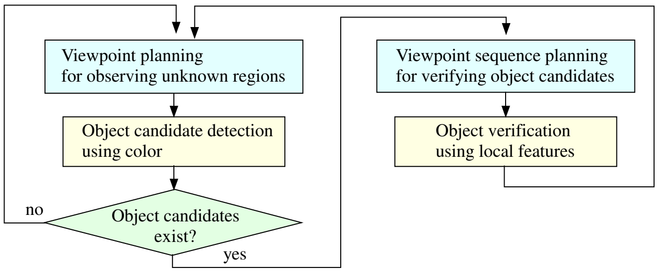



Fig. 2. Flow of two-stage observation planning and execution.

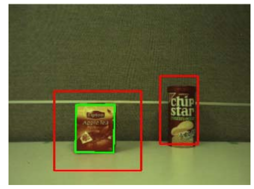


Fig. 3. Object recognition example.

object A object B object C

Fig. 4. Three objects used.

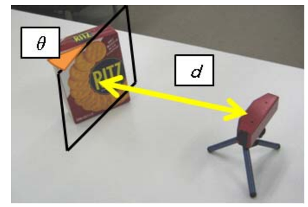
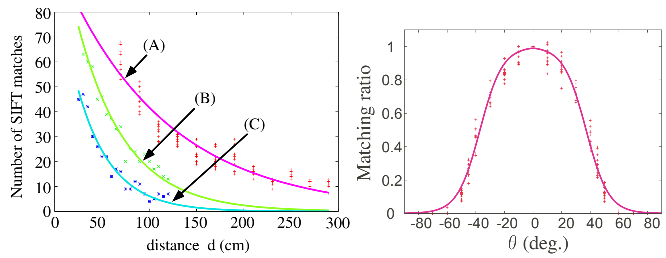



Fig. 5. Observation parameters for appearance modeling.




(a) Effect of distance. (b) Effect of angle.

Fig. 6. SIFT appearance modeling.

Shirai [14] have dealt with such a trade-off between visual data quality and vision cost in a route planning problem under uncertainty. In the scenario treated in this paper, a similar trade-off should be considered because there are multiple objects to detect in general.

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

## II. Overview of the Method

The environment information summarization treated in this paper is to (1) make a map of free space and (2) detect specified objects and record their positions in the map. We use the robot shown in Fig. 1 that has a range finder to detect free spaces and vision to detect objects; it can change the viewing direction by rotating the head. The robot is given a set of the appearance models of objects to detect.

Object detection is done in two stages (see Fig. 2). At the first stage, the robot detects object candidates using color information within the current visible areas. To explore unknown regions for candidate detection, an observation planning is performed to determine the next viewpoint. Once a set of object candidates is obtained, another observation planning for recognizing objects based on local features is performed to determine a sequence of viewpoints to verifying every candidate. The robot alternately performs these two kinds of observation planning and their execution until all candidates are detected and verified.

## III. Object Appearance Model and Object Recognition

### A. Visual features

Object recognition is done in two stages. In the first stage, object candidates are detected by using color. Color information of each object is modeled by a color histogram. An efficient search strategy [15] is used in this stage. In the second stage, each detected candidate is verified using local features. We use SIFT [16] as the feature and each object is modeled by a set of SIFT features extracted off-line. If an enough number of SIFT matches are obtained between the model of an object and the current input, the object is considered recognized. Fig. 3 shows an example of candidate detection and verification for object A shown in Fig. 4. Two objects are detected as candidates (red rectangles) and one of them is verified (green rectangle). Such a hierarchical use of features can reduce the cost for recognition [17].

### B. Appearance model for a surface

An object is composed of multiple surfaces. We here consider an appearance model for one surface. We use one color model for one object because color information is insensitive to scale changes. In the case of SIFT, however, although the feature itself is scale invariant to some extent, a large amount of scale change (i.e., change of distance to the object) affects the set of SIFT features to be observed. The number of SIFT matches usually decreases as the distance from the camera to the surface increases. The relative angle between the surface and the optical axis is another cause of decreasing matches. We model these effects.

Fig. 5 shows two parameters, distance $d$ and angle $\theta$ about the vertical axis, used for the modeling. Fig. 6(a) shows the relationships between the distance and the number of matches; we fit an exponential function independently to the data of each object. Fig. 6(b) shows the one between the angle and the number of matches normalized by the number when the angle is zero. Since the data shows the same tendency for all objects, we fit a single sigmoid curve to all the data.

Let $f$ and $g$ be the fitted functions for distance and for angle, respectively. Assuming that the effects of the distance and the angle are independent, the predicted number $\hat{z}$ of SIFT matches is given by the product of the two functions:

$$\begin{aligned}&\hat{z}(\boldsymbol{X}_c, \boldsymbol{X}_{obj}, m_i, \theta_i) \\ &\quad = f(dist(\boldsymbol{X}_c, \boldsymbol{X}_{obj}))\, g(angle(\boldsymbol{X}_c, \boldsymbol{X}_{obj}, \psi, \phi)), \quad (1)\end{aligned}$$

where $\boldsymbol{X}_c$ is the viewpoint, $\boldsymbol{X}_{obj}$ is the object position, $\psi$ is the surface orientation with respect to the object coordinates, $\phi$ is the orientation of the object around the vertical axis. $dist$ and $angle$ are the functions to calculate distance $d$ and relative angle $\theta$ shown in Fig. 5, respectively.

### C. Appearance model for an object

An object has in general different appearances for different surfaces. It is thus necessary to make multiple appearance models viewed from several directions. The more models are used, the more reliably an object is recognized but the more costly the recognition will be.

We use a fixed threshold $th_{match}$ on the number of SIFT matches for judging if an object is recognized. Currently, $th_{match}$ is set to ten. We thus chose four as the number $M$ of appearance models for one object; if the robot can comes to $50\,[cm]$ to an object, the number of matches is expected to exceed the threshold even when the angle is in the worst case, that is, $45\,[deg.]$.

### D. Object orientation estimation using appearance models

The effective distance for recognition of an object depends on what surface of the object is visible. So we on-line estimate the object orientation (i.e., $\phi$ in eq. (1)) using the appearance models.

We discretize the orientation into $N$ representative angles, $\phi_i\,(i = 1 \ldots N)$ (currently, $N = 8$). We assume the uniform distribution in the initial state. In one observation, we obtain a set of the numbers of SIFT matches $\boldsymbol{z} = \{z_1, \ldots, z_M\}$ between the SIFT features detected in the current search region and the $M$ models. The posterior probability of the model being at $\phi_i$ given observation $\boldsymbol{z}$ is estimated by:

$$P(\phi_i|\boldsymbol{z}, \boldsymbol{X}_c, \boldsymbol{X}_{obj}) = \alpha P(\boldsymbol{z}|\phi_i, \boldsymbol{X}_c, \boldsymbol{X}_{obj}) P(\phi_i), \qquad (2)$$

where $P(\phi_i)$ is the prior probability and $1/N$ at the initial state. $P(\boldsymbol{z}|\phi_i, \boldsymbol{X}_c, \boldsymbol{X}_{obj})$ is the likelihood function given by:

$$\begin{aligned}&P(\boldsymbol{z}|\phi_i, \boldsymbol{X}_c, \boldsymbol{X}_{obj}) \\ &\quad = \exp\left(-k \sum_{j=1}^{M} |z_j - \hat{z}(\boldsymbol{X}_c, \boldsymbol{X}_{obj}, \psi_j, \phi_i)|\right), \quad (3)\end{aligned}$$

where $k$ is a constant (currently, $k = 0.1$) and surface orientation $\psi_j$ is calculated from surface ID $j$.

### E. Predicted recognition probability

In observation planning, it is important to predict the probability of successful recognition for each observation. If we knew the orientation of the object, we could predict the number of SIFT matches using the corresponding appearance model and determine if the recognition succeeds. Since in reality, we can only determine the distribution of the orientation (see eq. (3)), however, we predict the probability of successful recognition $P_{recog}(\boldsymbol{X}_c, \boldsymbol{X}_{obj})$ as the one that the number of matches exceeds the threshold $th_{match}$:

$$P_{recog}(\boldsymbol{X}_c, \boldsymbol{X}_{obj}) = \sum_{i=1}^{N} P^i_{recog}(\boldsymbol{X}_c, \boldsymbol{X}_{obj}, \phi_i) \qquad (4)$$

$$\begin{aligned}&P^i_{recog}(\boldsymbol{X}_c, \boldsymbol{X}_{obj}, \phi_i) \\ &\quad = \begin{cases} P(\phi_i) & \exists\, j\, \{\hat{z}(\boldsymbol{X}_c, \boldsymbol{X}_{obj}, \psi_j, \phi_i) \geq th_{match}\} \\ 0 & \text{otherwise} \end{cases} \quad (5)\end{aligned}$$

This predicted probability is used for the observation planning described below.

## IV. Observation planning method

Two kinds of observation planning are performed corresponding to the two phases in object detection (see Fig. 2). The robot is given the shape and the size of the room for environment information summarization but not the placement of objects inside. It thus needs to actively explore the room to search for objects.

We use an occupancy grid map for geometric representation of the room. The cells of the map is classified into three categories: *free*, *obstacle*, and *unobserved*. Since the current summarization task requires detecting all specified objects, the robot must not leave a part of the room unobserved.

The robot plans a sequence of subgoals, at which it updates the occupancy map, detects object candidates, and makes a plan for verifying the object candidates. This section explains each step in detail. The planning for verification is further decomposed into two subproblems: determining the observation order of the candidates and determining the viewpoint sequence.

### A. Object candidate detection

In object candidate detection, the robot observes all unexamined regions visible from the current subgoal by changing its viewing direction. If a candidate is found by using color, the robot calculates its position using stereo and records it in the map. The range of the size of search window for an object is determined from the size of the object, which is known in advance, and the distance information obtained from the grid map.

### B. Subgoal planning for observing unknown regions

The next subgoal is selected in the currently-known free spaces so that the increase of the observed area divided by the movement cost is maximized. The utility function is thus defined as:

$$U_{subgoal}(\boldsymbol{X}) = \frac{\Delta I(\boldsymbol{X})}{C(\boldsymbol{X}, \boldsymbol{X}_c)}, \qquad (6)$$

where $\Delta I(\boldsymbol{X})$ is the predicted area of newly observed region by observing at $\boldsymbol{X}$ and $C(\boldsymbol{X}, \boldsymbol{X}_c)$ is the cost to move from the current position $\boldsymbol{X}_c$ to $\boldsymbol{X}$ (see Fig. 7).

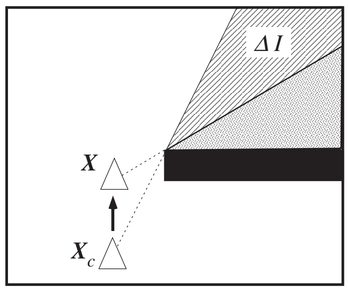



Fig. 7. Subgoal planning.

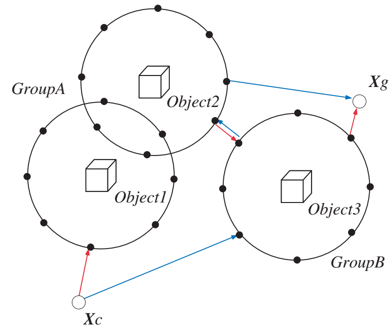



Fig. 8. Object grouping.

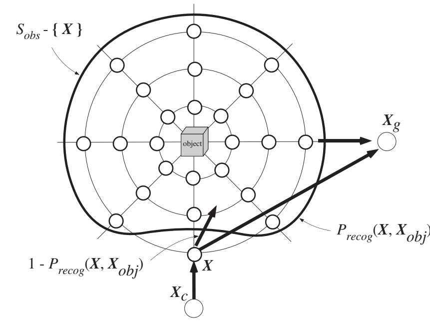



Fig. 9. One object verification.

### C. Object candidate grouping

If two or more objects exist in a small region, the robot can observe them from a single viewpoint. Such an observation is sometimes effective in reducing the traveling cost. We thus group objects which are near enough to each other. Each object has the maximum recognizable range $R_{max}$. If the distance between two objects is less than the summation of the maximum ranges of the objects, it is possible that the two objects can be verified at a single viewpoint. The object candidates are grouped which are connected with the simultaneously-observable links. In Fig. 8, for example, objects 1 and 2 forms a group and object 3 forms the other.

### D. Determining the observation order for groups

Suppose there are $N_g$ groups to observe. For an order of observation, $G_1, G_2, \ldots, G_{N_g}$, the total traveling cost is given by:

$$\begin{aligned} & C_{group}(\boldsymbol{X}_c, \boldsymbol{X}_g, G_1, \ldots, G_{N_g}) \\ & \quad = dist_{g-p}(G_1, \boldsymbol{X}_c) + \sum_{i=1}^{N_g-1} dist_{g-g}(G_i, G_{i+1}) \\ & \quad + dist_{g-p}(G_{N_g}, \boldsymbol{X}_g), \end{aligned} \tag{7}$$

where $dist_*$ is either the group-to-group or the group-to-point distance defined as:

$$dist_{g-p}(G, \boldsymbol{X}) = \min_{\boldsymbol{X}_G \in S_G} dist(\boldsymbol{X}_G, \boldsymbol{X}), \tag{8}$$

$$dist_{g-g}(G_i, G_j) = \min_{\boldsymbol{X}_{G_i} \in S_{G_i}} dist_{g-p}(G_j, \boldsymbol{X}_{G_i}), \tag{9}$$

$$S_G = \cup_k S_{obs_k}, \tag{10}$$

where $S_{obs_k}$ is the set of observation candidate points (explained below in Sec. IV-E) for the $k$th object candidate in group $G$.

The best order is currently found by an exhaustive search because the number of groups to be considered at a time is not large (usually two to three) in the current experimental setting. Some approximate methods would be necessary in a more complex environment.

In the case of Fig. 8, for example, there are two groups A and B and two orders (the blue and the red path, respectively) are possible. Since the red one is shorter, we first observe Group A and then Group B.

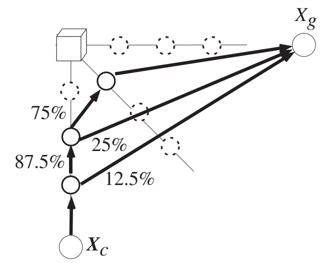



Fig. 10. One object verification plan example.

### E. Verification planning for one object

Each object candidate detected using color is verified using SIFT features. To verify a candidate, the robot needs to approach and observe it. Since one objective of the robot is to detect all specified objects, it has to make an observation plan for verification by considering the current position and the position for verifying the next candidate (or the next subgoal). Fig. 9 illustrates the case where the robot starts at $\boldsymbol{X}_c$, verifies an object, and move to the position $\boldsymbol{X}_g$.

Let $\boldsymbol{X}$ be the position for verification. We define the set $S_{obs}$ of possible positions for verification as the 24 points on three co-centric circles. The three radii are object specific, determine by considering the SIFT appearance model of the object (see Fig. 6).

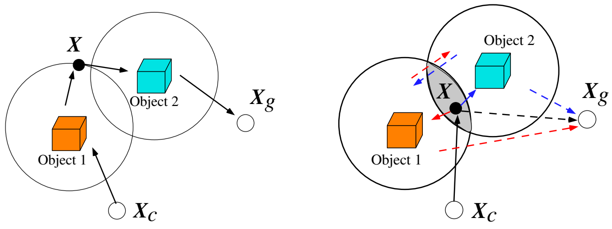
The criterion used for verification planning is to minimize the expected traveling distance from $\boldsymbol{X}_c$ to $\boldsymbol{X}_g$. The minimum of the expectation, $C^*_{obj}$ is given by:

$$\begin{aligned} & C^*_{obj}(\boldsymbol{X}_c, \boldsymbol{X}_g, S_{obs}) = \\ & \min_{\boldsymbol{X} \in S_{obs}} \left[ dist(\boldsymbol{X}_c, \boldsymbol{X}) + P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj}) dist(\boldsymbol{X}, \boldsymbol{X}_g) \right. \\ & \quad \left. + (1 - P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj})) C^*_{obj}(\boldsymbol{X}, \boldsymbol{X}_g, S_{obs} - \{\boldsymbol{X}\}) \right]. \end{aligned} \tag{11}$$

Fig. 10 shows an example verification plan for one object. The plan suggests three viewpoints to be tried subsequently; once the object is successfully verified, the robot immediately goes to the goal. The percentage of each edge of the graph is the probability of the robot taking that edge.

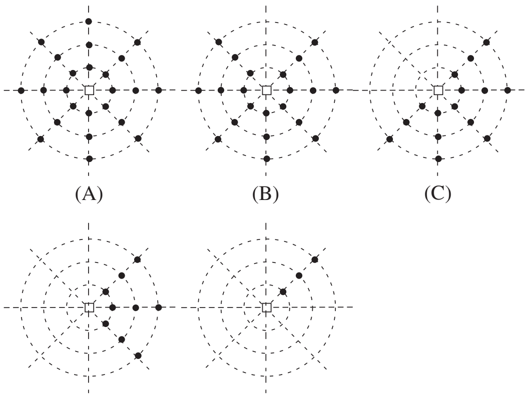
### F. Verification planning for multiple object candidates

When multiple object candidates are observable from some viewpoint, we can seek several observation strategies. Let us consider the case where there are two candidates. We have two strategies: sequential and parallel.




(a) Sequential strategy. (b) Parallel strategy.

Fig. 11. Two observation strategies.



(A) (B) (C)

(D) (E)

Fig. 12. Patterns of viewpoint set. The object is at the center and small black circles indicate possible viewpoints.

The sequential strategy observes two candidates one by one. Fig. 11(a) shows an example. The minimum of the total traveling cost is given by:

$$C^*_{seq}(\boldsymbol{X}_c, \boldsymbol{X}_g, \{Obj_1, Obj_2\}) = \min_{\boldsymbol{X} \in S_{obs_1}} \left[ C^*_{obj_1}(\boldsymbol{X}_c, \boldsymbol{X}, S_{obs_1}) + C^*_{obj_2}(\boldsymbol{X}, \boldsymbol{X}_g, S_{obs_2}) \right]. \quad (12)$$

This is actually the order for Objects 1 and 2; the cost can be estimated for the other order and the smaller one is selected.

A naive calculation of this equation is very costly because $C^*_{obs}$ should be calculated for many times with similar $\boldsymbol{X}$'s. We therefore make look-up tables (LUT's) for several patterns of viewpoint set. An LUT is represented as $C_{table}(\boldsymbol{X}_s, \boldsymbol{X}_g, Pat_j, \phi_i)$ with four parameters: start position, goal position, viewpoint pattern, and the object orientation. A viewpoint pattern indicates how possible viewpoints are distributed determined by the space configuration around the object. When an object is on a desk, for example, the robot may be able to approach it only from one direction. We currently use five predetermined patterns shown in Fig. 12 and the closest one is selected according to the actual space configuration. Since we estimate the distribution of object orientation every time a new observation is obtained, this distribution is used for cost calculation using LUT's accordingly.

The cost $\hat{C}_{obj}$ using LUT is given by:

$$\hat{C}^*_{obj}(\boldsymbol{X}_c, \boldsymbol{X}_g, S_{obs}) = \min_{\boldsymbol{X}_p, \boldsymbol{X}_q \in \hat{S}_{obs}} \left[ \begin{array}{l} dist(\boldsymbol{X}_c, \boldsymbol{X}_p) + dist(\boldsymbol{X}_q, \boldsymbol{X}_g) + \\ \sum_{i=1}^{N} C_{table}(\boldsymbol{X}_p, \boldsymbol{X}_q, Pat_{obs}, \phi_i) P(\phi_i) \end{array} \right]. \quad (13)$$

Using this cost, the cost for sequential observation (see eq. (12)) is approximated by:

$$\hat{C}^*_{seq}(\boldsymbol{X}_c, \boldsymbol{X}_g, \{Obj_1, Obj_2\}) = \min_{\boldsymbol{X} \in S_{obs_1}} \left[ \hat{C}^*_{obj_1}(\boldsymbol{X}_c, \boldsymbol{X}, S_{obs_1}) + \hat{C}^*_{obj_2}(\boldsymbol{X}, \boldsymbol{X}_g, S_{obs_2}) \right]. \quad (14)$$

The parallel strategy observes two object from one viewpoint. The region of possible such viewpoints is the intersection of two observable regions (see Fig. 11(b)). We discretize the intersection region to determine the set $S_{par}$ of viewpoints. For keeping the planning cost tractable, we set a limitation that a simultaneous observation of two objects is performed once for a pair of objects. If both objects are not verified, we switch to the sequential strategy. The cost of the parallel strategy is given by:

$$C^*_{par}(\boldsymbol{X}_c, \boldsymbol{X}_g, S_{par}, \{Obj_1, Obj_2\}) = \quad (15)$$

$$\min_{\boldsymbol{X} \in S_{par}} \left[ \begin{array}{l} dist(\boldsymbol{X}_c, \boldsymbol{X}) \\ + P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj_1}) P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj_2}) dist(\boldsymbol{X}, \boldsymbol{X}_g) \\ + (1 - P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj_1})) P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj_2}) \cdot \\ \qquad \hat{C}^*_{obj_1}(\boldsymbol{X}, \boldsymbol{X}_g, S_{obs_1}) \\ + P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj_1}))(1 - P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj_2})) \cdot \\ \qquad \hat{C}^*_{obj_2}(\boldsymbol{X}, \boldsymbol{X}_g, S_{obs_2}) \\ + (1 - P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj_1}))(1 - P_{recog}(\boldsymbol{X}, \boldsymbol{X}_{obj_2})) \cdot \\ \qquad \hat{C}^*_{seq}(\boldsymbol{X}, \boldsymbol{X}_g, \{Obj_1, Obj_2\}) \end{array} \right].$$

We compare the two strategies and select the better one. When more than two candidates exist in a group and if all candidates are observable from a single viewpoint, we conduct a similar calculation for the parallel strategy; that is, we try to verify all candidates simultaneously and if all are not verified, we take the sequential strategy.

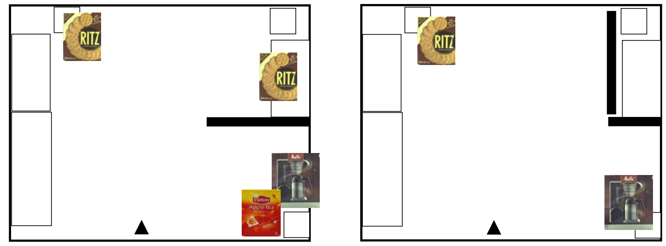
## V. Experimental Results

### A. Experimental setting

We show results of environment information summarization of a room of $7.0\,[m] \times 5.5\,[m]$. We use the robot shown in Fig. 1. The objects shown in Fig. 4 are put in the room but the number of objects is not given to the robot. We used two configurations shown in Fig. 13.

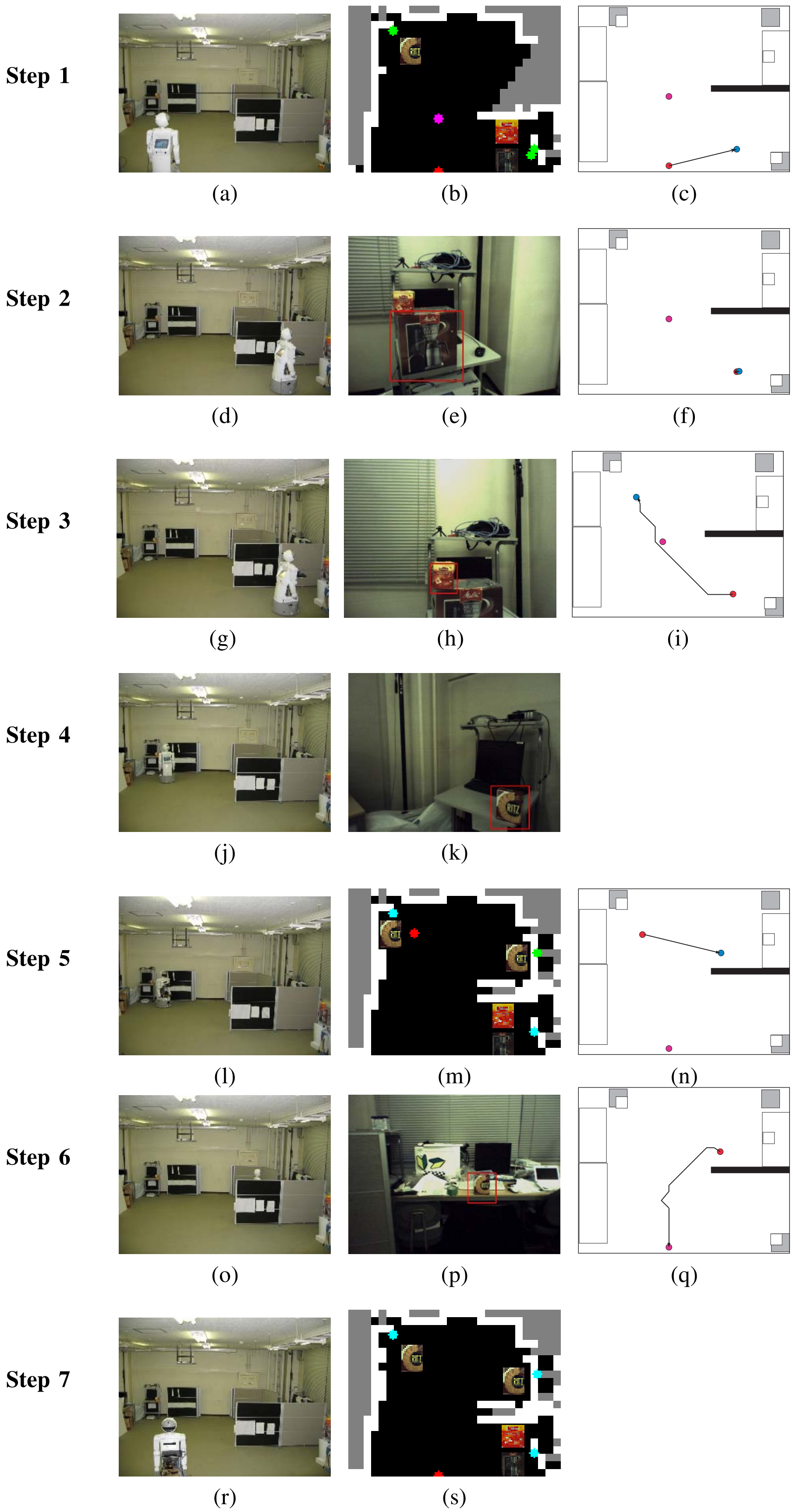
### B. First experiment

We use configuration (a) in Fig. 13. There are four objects to detect and two of them (objects A and C) are at almost the same position. The results of planning and execution are summarized in Fig. 14.




(a) (b)

Fig. 13. Two experimental settings. Small rectangles indicate desks and shelves. Black bold lines indicate partition walls which may occlude objects. Objects to detect are put on desks.




**(a)** observe the environment at the initial position (red mark). **(b)** made a free space map, found three object candidates (green marks), and determined the next subgoal (pink mark). **(c)** made a plan for verifying two candidate on the right.

**(d)** tried to verify two candidates. **(e)** successfully verified object C but failed to verify object A. **(f)** made a plan for verifying object A.

**(g)** moved to the planned verification position. **(h)** verified object A. **(i)** made a plan to verifying the remaining candidate (object B).

**(j)** moved to the candidate. **(k)** verified object B.

**(l)** observe the unknown region. **(m)** updated the free space map and found another candidate (green mark). The verified objects are indicated by blue marks. **(n)** made a plan for verifying object B.

**(o)** moved to the candidate. **(p)** verified object B. **(q)** made a plan to go back to the initial position because there remain no unknown region nor unverified candidates.

**(r)** came back to the initial position (i.e., the goal position). **(s)** the final summarization result.

Fig. 14. Experimental result for the configuration shown in Fig. 13(a).

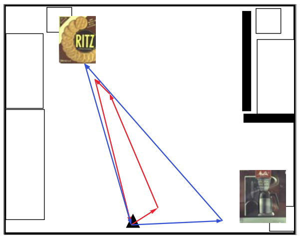



Fig. 15. Comparison results. The red and the blue path are by the planning with object appearance models and the one without models, respectively.

The process of environment information summarization is as follows:

**Step 1:** The robot observed the environment at the initial position, found three candidates, and made the first plan for verifying two of them.
**Step 2:** The robot tried to verify two objects simultaneously but failed to verify one of them.
**Step 3:** The robot verified the remaining one by moving closer to observe. After that, the robot made the plan for verifying the third candidate.
**Step 4:** The robot verified the object.
**Step 5:** The robot updated the map with a new candidate, and made a plan for verifying it.
**Step 6:** The robot verified it and made a plan to the goal position (i.e., the same as the initial position) because there remained no unknown regions nor unverified candidates.
**Step 7:** The robot finished the task and obtained the summarization result; it describes both the approximate shape of the room and the placement of the specified objects.

### C. Second experiment

We compared our method with a simple planning method which does not use object appearance models. Without appearance models, the robot does not know in what distance range an object can be recognized and, therefore, it moves to the closest viewpoint for the object. A comparison was made for the configuration shown in Fig. 13(b). The planned and executed paths for the methods is shown in Fig. 15. The traveling distances are $8.15\,[m]$ and $11.45\,[m]$ for the method using the appearance models and the one without the models, respectively. This shows the effectiveness of appearance model-based observation planning.

## VI. Conclusions and Future Work

This paper has described an observation planning method for environment information summarization. The planning problem includes subgoal planning for space exploration and candidate detection and viewpoint planning for object verification. The second planning problem is further decomposed into two subproblems: determining the observation order of the candidates and determining the viewpoint sequence. We developed object appearance models which represent the relationships between the observation condition and the recognizability. The models are used for considering the trade-off between the visual data quality and vision cost. We successfully applied the proposed method to actual environment summarization task using a humanoid robot with vision and range sensors.

Currently, the number of objects to detect is limited. We need to add models for more complex environments with various objects; adding visual features to use could be necessary to discriminate many objects. Improving the planning and the recognition algorithms is also important for more efficient summarization.

An interesting extension of the research is to consider time limit. The current planning algorithm makes a plan to explore the entire space and to verify all candidates detected. If the time for summarization is limited, it becomes necessary to choose regions to explore and objects to verify. This will be a challenging problem.


*Acknowledgment*

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


## 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] 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.

[4] 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.

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

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] P. Wang, R. Krishnamurti, and K. Gupta. View Planning Problem with Combined View and Traveling Cost. In *Proceedings of 2007 IEEE Int. Conf. on Robotics and Automation*, pp. 711–716, 2007.

[11] 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.

[12] K. Shubina. Visual Search for an Object in a 3D Environment Using a Mobile Robot. Technical Report CSE-2008-02, Department of Computer Science and Engineering, York University, April 2008.

[13] 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.

[14] J. Miura and Y. Shirai. Vision and Motion Planning for a Mobile Robot under Uncertainty. *Int. J. of Robotics Research*, Vol. 16, No. 6, pp. 806–825, 1997.

[15] V.V. Vinod and H. Murase. Image Retrieval Using Efficient Local-Area Matching. *Machine Vision and Applications*, Vol. 11, pp. 7–15, 1997.

[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] A. Shibata and J. Miura. Vision Planning for Object Search using Multiple Visual Features. In *Proceedings of 14th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, pp. 214–219, 2008.