



Active object search in an unknown large-scale environment using commonsense knowledge and spatial relations

Mingu Kim¹ · Il Hong Suh²

Received: 8 May 2018 / Accepted: 18 August 2019 / Published online: 29 August 2019
© Springer-Verlag GmbH Germany, part of Springer Nature 2019

Abstract

In this study, the goal is to efficiently and actively search for a target object in a previously unknown large-scale environment. To this end, we develop a probabilistic environment model that can utilize spatial commonsense knowledge and environment-specific spatial relations. The model evaluates the merit of exploring each possible viewpoint in the environment to find the target object. Then, the path planning method incorporates the estimated value of these viewpoints and the time cost between them to generate an efficient search path that minimizes the total search time. We also describe a search space reduction method that improves the feasibility of the proposed approach in large-scale environments. To validate the approach, we compare the search times of the proposed method to those of human participants, a coverage-based search and a random search in simulation experiments. The results show that the proposed method can generate search paths with similar search times to those of human participants, while clearly outperforming the coverage-based and random search methods. We also demonstrate the applicability of the approach in real-world experiments in which the robot could find the target object without a single failure case in 70 trials.

Keywords Active object search · Mobile robot · Probabilistic environment model

1 Introduction

Service robots perform everyday tasks such as fetch-and-carry tasks and mobile manipulation, which often involve physical interaction with various objects. For the robots to accomplish these tasks, they must first know where all the objects relevant to the given task are; however, it is unlikely that all task-related objects will already be within the robot's sensory range when the task is requested.

One way around this issue is to build an environment model that registers the locations of all objects in advance, and let the robot update object locations in the model when it detects changes [1,2]. However, this approach is impractical for dynamic environments, such as offices and homes, because the model can easily become fragile with frequent or simultaneous object movement by people in the environment. In addition, this method is not applicable when the robot must perform tasks in novel environments. Therefore, it is crucial for service robots to have the ability to actively search for objects required to complete their tasks.

The goal of active object search is to generate a sequence of sensing actions to localize a target object within the robot's field of view while minimizing the expected travel cost, here defined as total search time. Sensing actions can be defined in accordance with the robot's degree of freedom and its sensors. For example, a set of sensing actions can consist of navigation actions that change the robot's camera position, and in a broader sense, it can also include the choice of the object recognition algorithm for analyzing the input image and the various sensor parameters: pan-tilt angles, focus, focal length, etc.

This work was supported by the Korea Evaluation Institute of Industrial Technology (Grant Number 10080638).

✉ Mingu Kim
mgkim1013@hanyang.ac.kr; mgkim1013@gmail.com

Il Hong Suh
ihuh@hanyang.ac.kr

¹ Department of Electronics and Computer Engineering, Hanyang University, Room 613, R&D Building, 222 Wangsimni-ro, Seongdong-gu, Seoul 04763, Korea

² Department of Electronics and Computer Engineering, Hanyang University, Room 409, IT/BT Building, 222 Wangsimni-ro, Seongdong-gu, Seoul 04763, Korea

The most significant obstacle in active object search problems is that the search space for the target object grows exponentially with the number of possible sensing actions and the size of environment. Therefore, to efficiently locate an object in a large environment, we must devise a method to reduce or prioritize the search space, and then develop an adequate path planning method.

There has been increasing interest in active object search problems. In [3], Shubina and Tsotsos presented a two-step search strategy for exhaustive exploration in an unknown environment. Ma et al. [4] proposed a similar greedy search strategy, utilizing global and local search methods. The main drawback of such exhaustive search approaches is that they are not scalable to large environments because of the exponential growth of the search space, and they can also easily suffer from uncertainty in object recognition.

To this end, many researchers have investigated the use of spatial relations or semantic knowledge to prioritize the search space. Kollar and Roy [5] proposed a co-occurrence-based model to predict an object's location and perform a breadth-first search to plan a path to find the target object. Samadi et al. [6] investigated using the Web to localize objects by assuming that an object and its typical location are frequently found together in documents. Kunze et al. [7,8] described an approach based on qualitative spatial relations from symbolic representations and Gaussian mixture models, respectively. Zhang et al. [9,10] used answer set programming to represent domain knowledge for an object search and hierarchical partially observable Markov decision processes (POMDP) for sensing action selection. Veiga et al. [11] presented a framework for active object search that integrates a semantic map for knowledge representation and a POMDP for decision making. In [12], the authors investigated the relation between time and an object's position, and described a temporal persistence modeling algorithm to model and predict object locations based on sparse temporal observations. Kunze et al. [13] formulated the active object search problem as an orienteering problem with history-dependent rewards and proposed a stochastic view planning method to maximize the expected reward.

The above-mentioned approaches have shown that by prioritizing the search area, a robot can find objects more efficiently compared to the exhaustive search methods. However, all of these previous approaches have the disadvantage of requiring a pre-built map of the search environment. One of the limitations of such methods is that they are not applicable to previously unknown environments. One might argue that a robot can explore the environment and build a map before they start to search object. However, such a map building process requires exploration of the entire environment, which could be quite time-consuming, and it does not make sense that the robot could not find the object while exploring and building a map. Therefore, the ability for a robot to

search a target object without a priori map can be crucial in such environments.

In more recent studies, Zhu et al. [14] proposed a deep reinforcement learning-based framework for the object search without pre-built map. They used deep network to map an observed image and a target image directly to the robot action. Although they showed the approach can be generalized to new targets and scenes, it required millions of training images. Also, the evaluation was done in a small-scale environment (a single room). Hanheide et al. [15] presented a framework to plan robot tasks in unknown environments, and it was evaluated in five tasks including object search. They used three-layered knowledge schema to represent task relevant instance, default, and diagnostic knowledge, and assumptive planning architecture composed with two planners: a classical planner and a decision theoretic planner (POMDP). The main difference between ours and Hanheide et al. is that we directly modeled the probability that each viewpoint contains the target object, while Hanheide et al. modeled the probability that each room contains the target and performed an exhaustive coverage-based search using POMDP after arriving in the room. In addition, we considered the mutual information (information gain) as well as the probability of a possible location of an object for path planning to find the target.

In this paper, we propose an effective approach for active object search in unknown large-scale environments. This approach uses a probabilistic graphical model to represent commonsense knowledge based on object–object, object–room, and room–room co-occurrences to overcome the limitations of exhaustive search-based approaches. As a way to deal with object search in unknown environment, the model is built incrementally as the robot searches for the target and obtains new input data. The contributions of this work are fourfold:

- We propose a search space reduction method that makes it feasible to solve object search problems in large-scale environments with a reasonable computational cost by decreasing the number of candidate viewpoints.
- We present a probabilistic environment model for object search that evaluates viewpoints in the environment by utilizing spatial commonsense knowledge and environment-specific spatial relations.
- We adopt an ant colony optimization (ACO) algorithm for path planning to minimize the total search cost while incorporating priority information about viewpoints.
- We evaluate the validity of the proposed approach in both simulation and real-world experiments.

The rest of this paper is organized as follows. In Sect. 2, we provide an overview of the proposed approach. Section 3 presents a method to reduce the number of viewpoints

that have to be considered. Section 4 describes the details of the probabilistic environment model and value estimation of each viewpoint. In Sect. 5, we show how to generate a path that minimize the total cost of the object search process. Finally, we present experimental results and draw final conclusion in Sects. 6 and 7, respectively.

2 Approach overview

In this study, we assume that the robot has object recognition and simultaneous localization and mapping (SLAM) capabilities, and is equipped with the necessary sensors, such as RGB-D sensors and odometers. Under these assumptions, the goal is to produce an efficient and optimized path for finding a target object in unknown large-scale environments.

We first gather new sensor data from the robot’s current view and build a partial grid map based on the new and previously gathered data. If there is no pre-gathered data, the map is built based only on the current sensor data. Then, the map is used to extract key positions and the connectivity information between them to reduce the search space. Throughout this paper, the term “node” is used to refer to these key positions, and how these key positions are extracted is described in the next section.

These nodes and their connectivity information is used to specify the global structure of a probabilistic environment model. Each node correspond to each local structure of the model which is designed to represent object–object, object–room, and room–room relation in category level. More details about the local structure can be found in Sect. 4. The connectivity information between nodes is used to relate the local structures. If two nodes are connected, corresponding two local structures are connected by undirected edges representing symmetric relations. While the global model structure is built based on current search environment, the model parameters are pre-trained from the training data. We used this model to estimate the value of viewpoints in the nodes based on obtained object recognition results, while the robot is searching for the target.

With the estimated values, a path planner generates the path that minimizes the expected time cost for the object search. Finally, as the robot follows the generated path, the process is repeated as new sensor data become available. Figure 1 illustrates an overview of the proposed object search process.

3 Reducing the search space

To deal with an active object search problem in a large-scale environment, a method to reduce the search space is imperative for reducing computational costs and search times. To

this end, we use several morphological image processing algorithms to obtain a node map that represents key positions of the candidate viewpoints and the connections between them. The search space reduction steps are described in Fig. 2.

First, since the occupancy grid map obtained from the SLAM process does not consider the robot’s dimensions, it is inflated by the robot’s radius to obtain an obstacle-free space the robot can occupy (Fig. 2a, b). Then, the inflated map is skeletonized by adding occupied points on the boundaries of the free space, but not allowing the space to be split (Fig. 2b, c). Finally, the node map is created by extracting end and branch points from the skeleton map, while adding waypoints between connected nodes if they are too far away from each other (Fig. 2c, d).

Due to space limitations, detailed algorithms have been omitted from this paper. Interested readers may refer to [16, 17] for further detail.

4 Estimating the value of viewpoints

In this work, we define the value of a viewpoint as the sum of the probability of the existence E_{t,v_i} of target object t in a scene from viewpoint v_i and the average amount of information about the existence $E_{t,v}$ of the target object in a scene from viewpoint v that can be obtained by an observation O_{c,v_i} of an object of category c from viewpoint v_i :

$$\text{Value}(v_i) = p(E_{t,v_i} = \text{true}) + \sum_{v \in V} \sum_{c \in C} \frac{I(E_{t,v}; O_{c,v_i})}{N_V N_C}, \quad (1)$$

where V and C represent a set of all viewpoints on the current node map and a set of all object categories in the environment, and N_V and N_C are the number of viewpoints and the number of all categories of objects, respectively. Here, the amount of information $I(E_{t,v}; O_{c,v_i})$ is defined as mutual information (MI):

$$\begin{aligned} I(E_{t,v}; O_{c,v_i}) &= \sum_{e_{t,v} \in E_{t,v}} \sum_{o_{c,v_i} \in O_{c,v_i}} p(e_{t,v}, o_{c,v_i}) \log \left(\frac{p(e_{t,v}, o_{c,v_i})}{p(e_{t,v})p(o_{c,v_i})} \right). \end{aligned} \quad (2)$$

Using the MI [18], we can measure the amount of uncertainty in the existence $E_{t,v}$ of target object t at viewpoint v that can be reduced by observation O_{c,v_i} of an object of category c from viewpoint v_i . For example, if a robot finds toothpaste while looking for a toothbrush, the average MI is increased, because the probability of finding the toothbrush at viewpoints near the place where the toothpaste was found increases, whereas the probability of other viewpoints

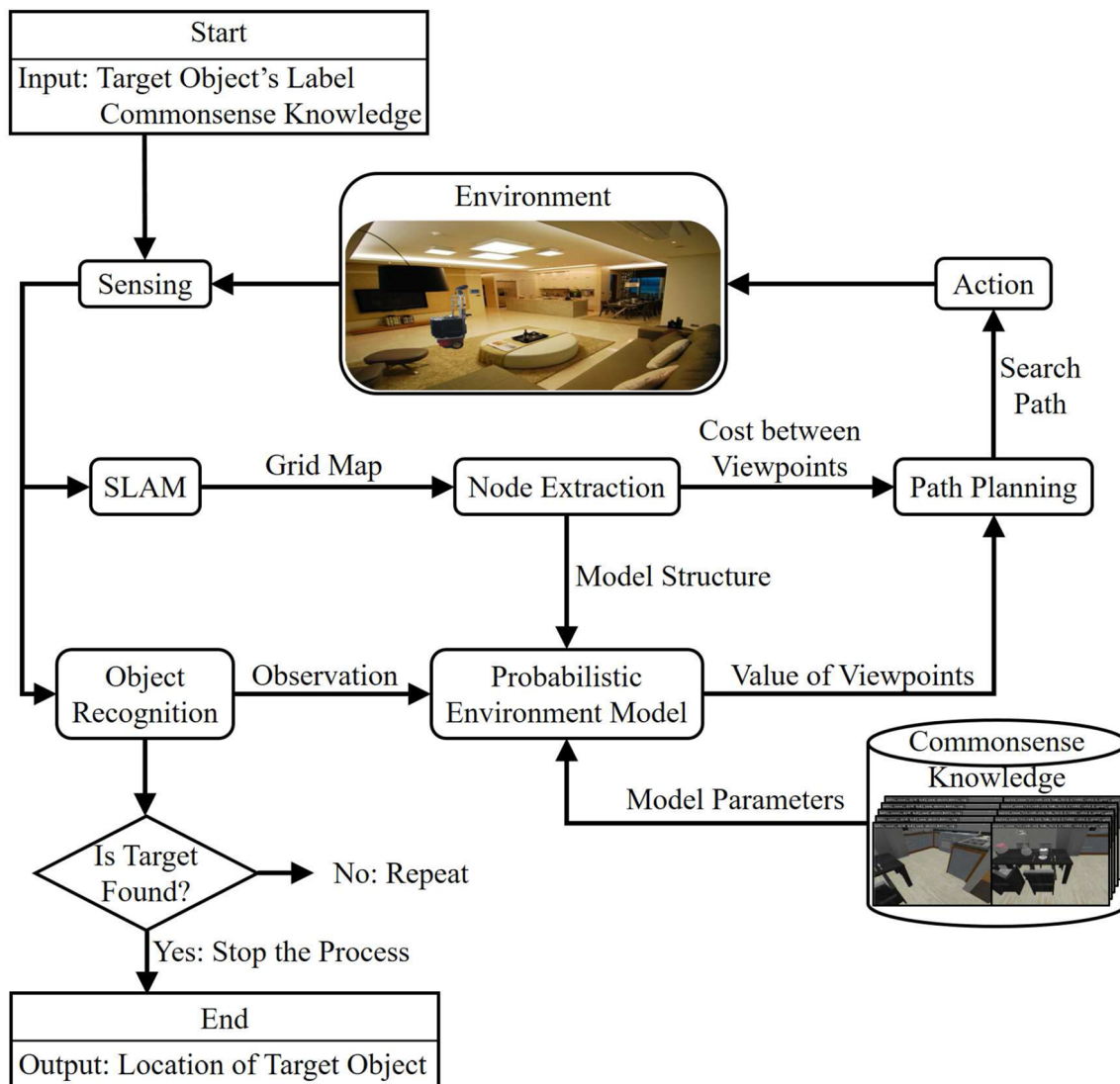


Fig. 1 Pipeline of the proposed active object search process

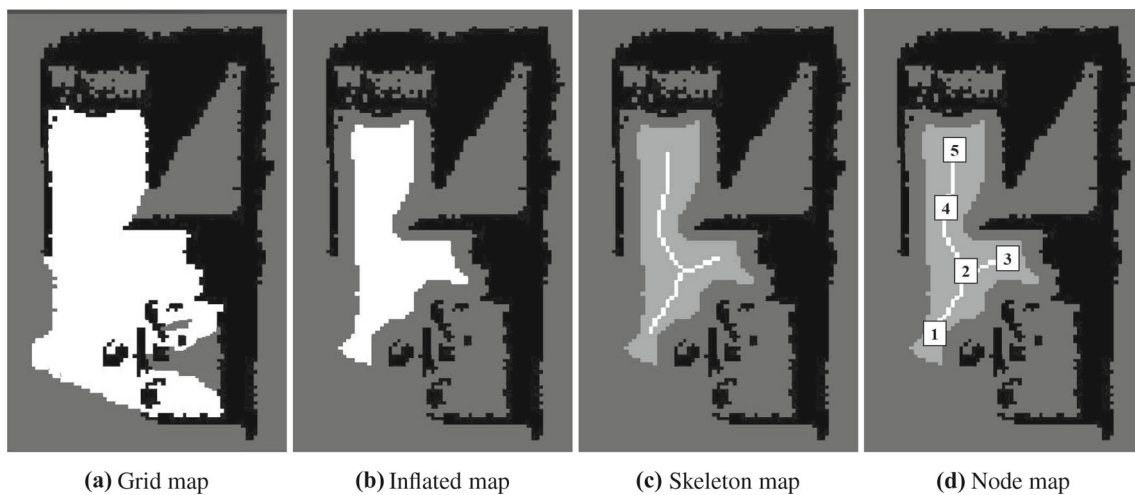


Fig. 2 Node extraction steps for search space reduction using morphological image processing

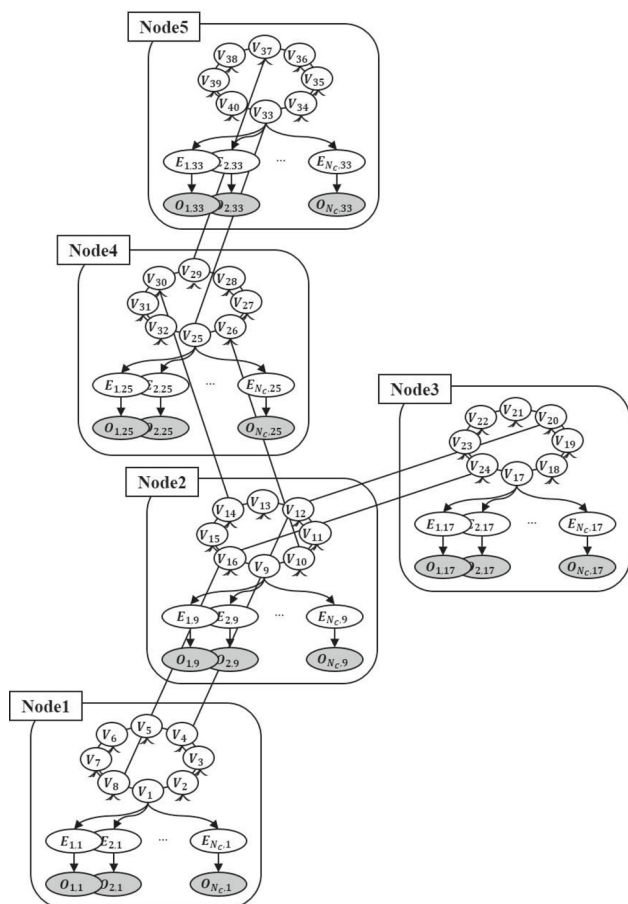


Fig. 3 Structure of the probabilistic environment model. The vertices represent random variables: viewpoint random variables V_i , object existence variables E_{c,v_i} , and observation variables O_{c,v_i} . Shaded vertices indicate that the values of these variables can be directly obtained from the observation, while unshaded vertices are hidden and can only be inferred based on the observations. The variables are grouped by rounded rectangles to describe the relationships between the node map and model’s global structure. The edges represent relationships between random variables, where directed edges represent causal relations, and undirected edges represent symmetric relations. The term “symmetric relation” is used to refer to a relationship that is not causal. More formally, $P(V_i, V_j) = P(V_j, V_i)$, where $P(E_{c,v_i} | V_i) \neq P(V_i | E_{c,v_i})$. Therefore, the relation of two viewpoints’ category is symmetric, while the relation between existence variable and viewpoint variables is not symmetric

decreases and uncertainty about the location of the toothbrush is reduced. The average MI is also increased if the robot finds a pillow or coffeemaker, which reduces the probability of finding a toothbrush near the location where these objects are found. As shown in this example, it is reasonable to add the average MI as well as the probability of finding the target object in a calculation of viewpoint value because the total cost of the search can be greatly reduced by letting a robot observe viewpoints that are likely to contain objects strongly related to the target object.

To calculate a viewpoint’s value, probability distributions $P(E_{t,v_i})$ and $P(E_{t,v}, O_{c,v})$ are necessary. To this end, we apply the chain graph model, which is a generalized probabilistic graphical model [19] that can represent both the causal relations of a Bayesian network and symmetric relations of a Markov random field.

To build a chain graph model for active object search, we utilize commonsense spatial knowledge that is not environment-specific. First, each node in the node map is discretized into eight viewpoints—one at every 45 degrees. The number of viewpoints per node can affect the computation time. The smaller the number, the better. The minimum number is, however, bounded according to the camera’s angle of view. In our case, it was 45 degrees, and we used 8 viewpoints per node. Viewpoint random variables V_i in the model represent the category of the scene from corresponding viewpoints (kitchen, bedroom, study, etc.), and each has a causal relation with the N_C binary random variables $E_{c,v}$. These existence variables represent whether an object of category c exists in viewpoint v . We also add an additional binary random variable, observation variable $O_{c,v}$, for each existence variable to account for false positive and negative results from imperfect object recognition. Finally, a global chain graph structure (including the number of nodes and the connectivity between them) is defined by utilizing the node map built in Sect. 3, which contains environment-specific spatial relations. Each group of variables corresponding each node in the node map is connected to each others, if there is direct path between the nodes. Figure 3 shows the structure of the chain graph model corresponding to the node map in Fig. 2d.

The 3D simulation environments in Fig. 5 are used to obtain the model parameters $P(V_i)$, $P(V_i, V_j)$, $P(E_{c,v_i} | V_i)$, and $P(O_{c,v_i} | E_{c,v_i})$. We collect 300 pairs of training images from each environment by randomly positioning the robot in the environment and moving it to adjacent viewpoints. Then, the training images are labeled to obtain the parameters, by counting the occurrences and co-occurrences of each label.

In the experiments in Sect. 6, a Markov chain Monte Carlo method with a Suwa–Todo sampler [20] is used to infer probability distributions $P(E_{t,v_i})$ and $P(E_{t,v}, O_{c,v})$ and estimate a viewpoint’s value. We compared two sampling methods, Suwa–Todo and Gibbs sampling, and Suwa–Todo sampler converged more rapidly than the Gibbs sampler.

5 Path planning for object search

In this section, we describe a path planning method that incorporates viewpoint values using an ACO algorithm [21]. ACO is a probabilistic approach to obtaining a near-optimal solution at relatively low computational cost; for that reason, it has been applied to a wide range of computationally expensive problems, such as the traveling salesman, job-shop

scheduling, and vehicle routing problems. In this study, we adopted the max–min ant system [22] variant of the ACO algorithm because its structure is relatively simple to implement.

An artificial ant in the proposed approach builds a candidate object search path by iteratively and stochastically selecting a next viewpoint based on the attractiveness η_{v_i, v_j} of reachable viewpoints from the current viewpoint while exploiting pheromone information τ_{v_i, v_j} accumulated on edges between viewpoints from previous ants. More formally, the probability p_{v_i, v_j} of selecting a next viewpoint v_j from a set of adjacent viewpoints V_{adj} from current viewpoint v_i is

$$p_{v_i, v_j} = \frac{(\tau_{v_i, v_j}^\alpha)(\eta_{v_i, v_j}^\beta)}{\sum_{v_k \in V_{\text{adj}}} (\tau_{v_i, v_k}^\alpha)(\eta_{v_i, v_k}^\beta)}, \quad (3)$$

where parameters α and β determine the influence of τ_{v_i, v_j} and η_{v_i, v_j} . We defined the attractiveness η_{v_i, v_j} of a transition between viewpoints v_i and v_j as

$$\eta_{v_i, v_j} = \frac{\text{Value}(v_j)}{\text{Time}(v_i, v_j)}, \quad (4)$$

where the time cost $\text{Time}(v_i, v_j)$ between viewpoints is calculated from the node map based on the linear and angular velocities of the robot. For example, if the angle difference and distance between two viewpoints are 45 degrees and 2 meters, respectively, the time cost of the robot with an angular velocity of 45 degrees/second and a linear velocity of 1 m/s is 3 s.

When every set of ants for an iteration has built a search path, the pheromone trails τ_{v_i, v_j} are updated by

$$\tau_{v_i, v_j} \leftarrow \left[(1 - \rho) \cdot \tau_{v_i, v_j} + \Delta\tau_{v_i, v_j}^{\text{best}} \right]_{\tau_{\min}}^{\tau_{\max}}, \quad (5)$$

where ρ is the pheromone evaporation coefficient; τ_{\max} and τ_{\min} are, respectively, the upper and lower bounds of the pheromone trails; and $\Delta\tau_{v_i, v_j}^{\text{best}}$ is the amount of pheromone deposited by the iteration. The amount of newly deposited pheromone $\Delta\tau_{v_i, v_j}^{\text{best}}$ is proportional to the quality of the best path in an iteration, where the quality of the path $Q(\text{path})$ is defined as

$$Q(\text{path} = (v_1, v_2, \dots, v_n)) = \frac{\sum_{v_j \in \text{path}} \text{Value}(v_j)}{\sum_{(v_i, v_j) \in \text{path}} \text{Time}(v_i, v_j)}. \quad (6)$$

As artificial ants repeatedly generate candidate paths and update pheromone trails, more pheromones are deposited between viewpoints of higher-quality search paths. That is,

the pheromone reflects the colony's accumulated experience while exploring. Since ants prefer to follow paths with more pheromone, the algorithm converges to a high-quality search path that visits viewpoints with higher values in less time than the other candidate paths.

6 Experiments

In this section, we show that the proposed approach can effectively search for a target object in an unknown large-scale environment. It is generally very difficult to compare performances of different object search approaches for several reasons. First, different object search algorithms were developed targeting different environments (previously known/unknown, large/small, home/office, etc.), and therefore, it is hard to say which algorithm is the state of the art. Also, there is no open dataset for fair comparisons between different object search algorithms. Therefore, we evaluate the performance of the proposed approach in simulation environments in comparison with human performance, coverage-based search, and random search as a baseline. We also evaluate it in a real environment to show the validity of the approach. For each trial in both experiments, the robot's probabilistic environment model and grid map were re-initialized before the search began to make sure the robot has no prior information about the environment.

6.1 Simulation experiment

We conducted extensive simulation experiments to quantitatively verify that the proposed object search approach can generate low-cost paths to find a target object in unknown large-scale environments. Simulation experiments took place in 60 different environments that are generated from five different house structures by changing the locations of objects in each structure. The size of 5 environments was about 13.5 m \times 9 m, and the size of one environment was about 22 m \times 14.5 m. The houses consisted of 7 room categories: kitchen, dining room, living room, dressing room, bedroom, study, and bathroom.

The simulated mobile robot was equipped with two RGB-D sensors: one, for object recognition, mounted on the head of the robot and one for localization and mapping, mounted on the base of the robot. The robot also used odometer data from the simulator, to which we added zero-mean Gaussian noise with a standard deviation of 5 cm for additional realism.

For object recognition in the simulation experiments, we used the logical camera module provided in the simulator by adding false negative and positive rates of 5% and 1%, respectively. For localization and mapping, we used real-time appearance-based mapping (RTAB-Map), which is an

Table 1 Comparison of average search times in simulation environments

	Human	Proposed	Coverage	Random
Avg. time (s)	17.2194	19.9864	180.8731	764.4873
SD	15.2943	16.4699	248.7436	4018.2940

Table 2 Comparison of average searched area in simulation environments

	Human	Proposed	Coverage	Random
Avg. area (m ²)	26.7581	28.6844	315.9765	1365.6848
SD	25.0107	26.3758	467.9845	6654.6784

open-source algorithm using an RGB-D graph-based SLAM approach [23].

To evaluate the proposed method, we compared the method to two baseline methods: a coverage-based search method based on [3] and a random search method. The random search method randomly selects a sensing action: turning left, turning right, or moving forward. We also compared the proposed method to human participants by letting 10 participants perform the object search tasks in the simulation environments using a first-person view and a grid map constructed while navigating. A total of 600 independent trials were performed for each approach. In each trial, the initial robot location and target object were randomly selected from the robot's free space in the environment and from 97 objects from 40 categories in the environment, respectively.

We recorded the total search time and searched area in each trial; Table 1 shows the average and standard deviation of the search times. As expected, humans found the target objects the most quickly. However, the difference in the average search time between the human participants and the proposed approach was only 2.7 s; the proposed method performed much better than the coverage-based search method and random search method. We also note that some of the differences in time between the human participants and the robot is caused by humans using the experiences about house structure from previous runs. Although we shuffled the environments to ensure the same structure cannot come out repeatedly, humans could easily use the previous experiences, while the robot always had no prior information about the environments. Table 2 shows the average and standard deviation of the searched area. It also shows the proposed approach is superior to the two baseline methods. We also note that two baselines searched the area that exceeds the total area of the given environment in some trials because of the uncertainty in object recognition.

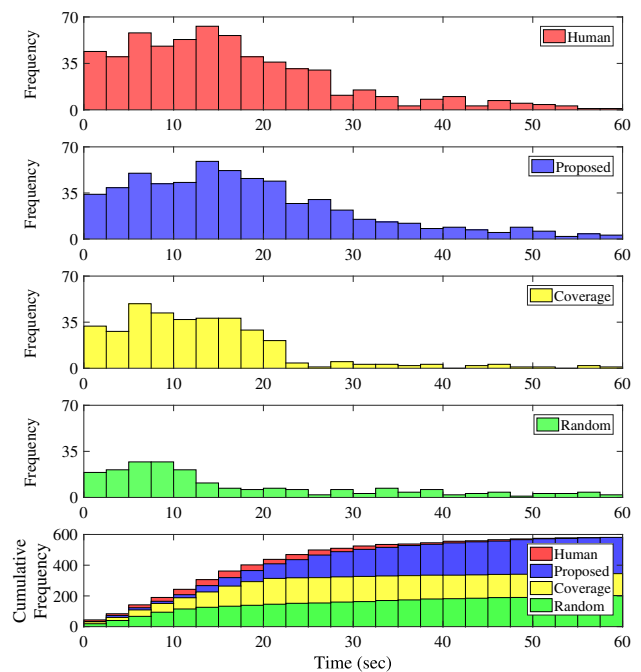
**Fig. 4** Frequency histograms of the simulation experiment results from human participants, the proposed approach, and a random search method

Figure 4 shows frequency histograms of the search time for all three cases. The distribution of the search time histogram of the proposed approach was similar to that of human participants, and clearly outperformed two baselines. The random search method also found target objects within 15 s for 25% of the search trials. However, the search time was highly dependent on the initial location of the robot and the location of the target object. That is, if the robot and target were in the same room, the robot could easily find the target using random search; however, in the worst case when the target was not in the same room, the search took about 5 hours. The performance of the coverage-based method was also highly dependent on the initial robot and target location, but its variance and maximum search time was much lower than those of random search.

To compare the search paths produced by the proposed method and human participants, we also conducted additional experiments by setting the same initial robot locations and target objects. Figure 5 shows the paths created by the proposed approach in these additional experiments, and a video recording is available online at <https://youtu.be/Cj5JvOaAw1s>. Note that throughout this experiment, the robot turns more frequently than the human participants, and is also more likely to enter a room to observe objects. These phenomena are believed result from the absence of a place recognizer. In the current system, the robot must observe objects to infer the category of a scene, but participants could more quickly classify a scene by observing its furniture or floor tiles.



Fig. 5 Example search paths generated by the proposed approach from different object search trials. The green circles correspond to the robot's initial locations, and the blue circles correspond to its final locations. The red circles indicate the locations of target objects (color figure online)

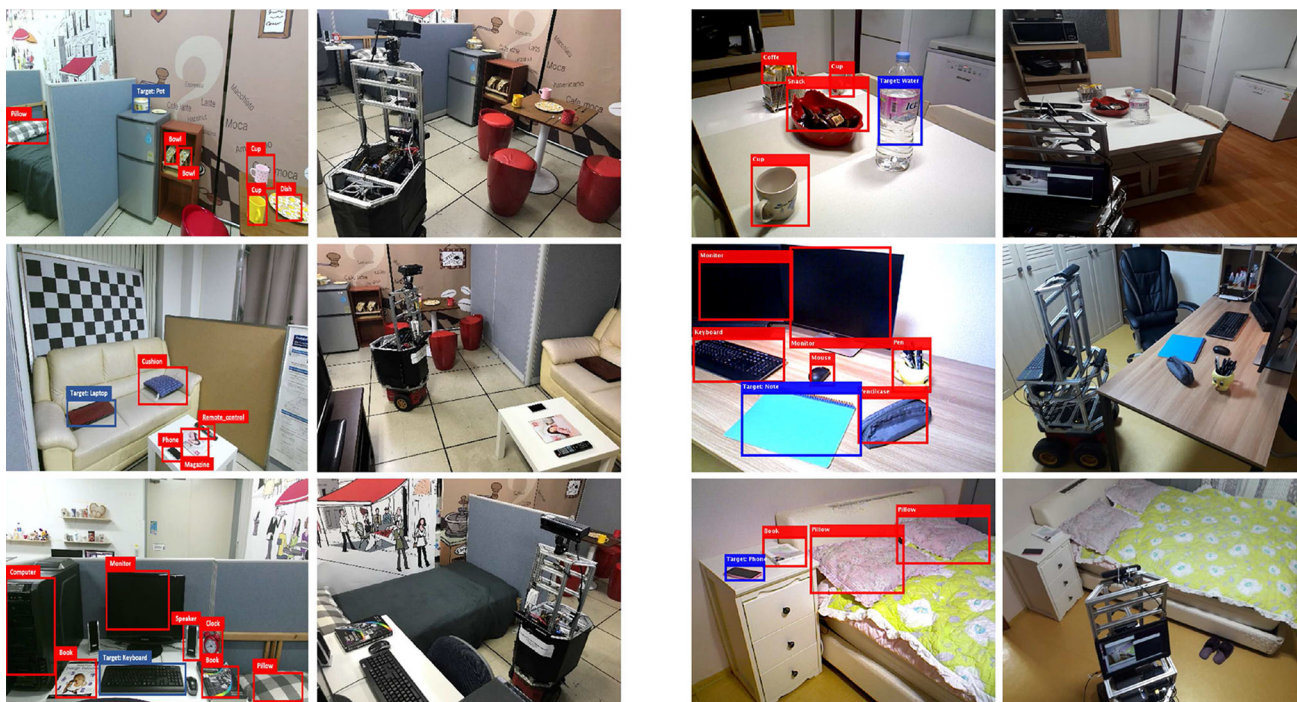


Fig. 6 Robot and environment used in the robot experiments

6.2 Robot experiment

We also conducted real robot experiments to verify the validity of the proposed approach in a real environment. The

experiments were performed in a 4.1 m × 5.5 m test bed and real environment of about 92 m². We used a Pioneer mobile robot with two RGB-D sensors, as shown in Fig. 6.

We used the same SLAM algorithm, RTAB-Map, as in previous experiments and chose an object recognizer that uses deep learning and a convolutional neural network (CNN) [24]. The object recognizer is pre-trained using the CIFAR-10 dataset, which has 50,000 training images; we fine-tuned the CNN model using 900 labeled images of 60 different target objects to increase its reliability.

Throughout the experiments, the robot successfully found the target objects in all 40 trials in test bed environment and all 30 trials in real environment without prior information about the environment. The average search time was 23.35 s and 62.68 s in the test bed and real environment, respectively, where the angular and linear velocities of the robot were set to 20 degrees/second and 0.15 m/s. The computational time per iteration of each module was 0.05 s for object recognition, 0.02 s for SLAM, 0.05 s for inference on the environment model, and 0.38 s for ant colony optimization for search path planning, on average.

A video of all 40 trials in the test bed environment can be found at <https://youtu.be/rpLZzmGaXW0>.

7 Conclusions and future work

In conclusion, we have proposed an efficient approach to active object search in unknown large-scale environments. The proposed approach utilizes commonsense spatial knowledge and environment-specific spatial relations based on the probabilistic environment model. We have also presented a search space reduction method that uses morphological image processing and an ACO-based path planning method that incorporates the value of viewpoints while minimizing the total search time. What sets our approach aside from most previous works is that we use no prior environment-specific information.

We have demonstrated the promising performance of our active object search approach in unknown large-scale environments with a large series of simulation experiments, and show that the approach can generate low-cost search paths that are only 2.7 s longer than those produced by human participants. In comparison with a random and coverage-based search, the proposed approach can provide significantly lower-cost paths. We have also shown the feasibility of applying the approach to a real-world environment in real robot experiments.

In future work, we plan to apply an incremental parameter learning method to the probabilistic environment model, so that the robot can find target objects more quickly in familiar environments. In addition, we also would like to extend the proposed approach by incorporating a place recognition algorithm that gives direct observations to viewpoint random variables to further improve the performance of the approach.

References

1. Suh IH, Lim GH, Hwang W, Suh H, Choi JH, Park YT (2007) Ontology-based multi-layered robot knowledge framework (OMRKF) for robot intelligence. In: 2007 IEEE/RSJ international conference on intelligent robots and systems (IROS), pp 429–436
2. Lim GH, Suh IH (2012) Improvisational goal-oriented action recommendation under incomplete knowledge base. In: 2012 IEEE international conference on robotics and automation (ICRA), pp 896–903
3. Shubina K, Tsotsos JK (2010) Visual search for an object in a 3D environment using a mobile robot. *Comput Vis Image Underst* 114(5):535
4. Ma J, Chung TH, Burdick J (2011) A probabilistic framework for object search with 6-DOF pose estimation. *Int J Robot Res* 30(10):1209
5. Kollar T, Roy N (2009) Utilizing object-object and object-scene context when planning to find things. In: 2009 IEEE international conference on robotics and automation (ICRA), pp 2168–2173
6. Samadi M, Kollar T, Veloso M (2012) Using the web to interactively learn to find objects. In: Twenty-Sixth AAAI conference on artificial intelligence, pp 2074–2080
7. Kunze L, Burbridge C, Hawes N (2014) Bootstrapping probabilistic models of qualitative spatial relations for active visual object search. In: 2014 AAAI spring symposium series, pp 24–26
8. Kunze L, Doreswamy KK, Hawes N (2014) Using qualitative spatial relations for indirect object search. In: 2014 IEEE international conference on robotics and automation (ICRA), pp 163–168
9. Zhang S, Stone P (2015) CORPP: Commonsense reasoning and probabilistic planning, as applied to dialog with a mobile robot. In: Twenty-Ninth AAAI conference on artificial intelligence, pp 1394–1400
10. Zhang S, Sridharan M, Wyatt JL (2015) Mixed logical inference and probabilistic planning for robots in unreliable worlds. *IEEE Trans Robot* 31(3):699
11. Veiga TS, Miraldo P, Ventura R, Lima PU (2016) Efficient object search for mobile robots in dynamic environments: semantic map as an input for the decision maker. In: 2016 IEEE/RSJ international conference on intelligent robots and systems (IROS), pp 2745–2750
12. Toris R, Chernova S (2017) Temporal persistence modeling for object search. In: 2017 IEEE international conference on robotics and automation (ICRA), pp 3215–3222
13. Kunze L, Sridharan M, Dimitrakakis C, Wyatt J (2017) Adaptive sampling-based view planning under time constraints. In: 2017 European conference on mobile robots (ECMR), pp 1–6
14. Zhu Y, Mottaghi R, Kolve E, Lim JJ, Gupta A, Fei-Fei L, Farhadi A (2017) Target-driven visual navigation in indoor scenes using deep reinforcement learning. In: 2017 IEEE international conference on robotics and automation (ICRA), pp 3357–3364
15. Hanheide M, Göbelbecker M, Horn GS, Pronobis A, Sjö K, Aydemir A, Jensfelt P, Gretton C, Dearden R, Janicek M et al (2017) Robot task planning and explanation in open and uncertain worlds. *Artif Intell* 247:119
16. Dougherty ER, Lotufo RA (2003) Hands-on morphological image processing, vol 59. SPIE Press, Bellingham
17. Vogt P, Riitters KH, Estreguil C, Kozak J, Wade TG, Wickham JD (2007) Mapping spatial patterns with morphological image processing. *Landscape Ecol* 22(2):171
18. Cover TM, Thomas JA (2012) Elements of information theory. Wiley, New York
19. Lauritzen SL, Richardson TS (2002) Chain graph models and their causal interpretations. *J R Stat Soc Ser B (Stat Methodol)* 64(3):321
20. Suwa H, Todo S (2010) Markov chain Monte Carlo method without detailed balance. *Phys Rev Lett* 105(12):120603

21. Dorigo M, Birattari M, Stutzle T (2006) Ant colony optimization. *IEEE Comput Intell Mag* 1(4):28
22. Stützle T, Hoos HH (2000) MAX-MIN ant system. *Future Generat Comput Syst* 16(8):889
23. Labbé M, Michaud F (2014) Online global loop closure detection for large-scale multi-session graph-based slam. In: 2014 IEEE/RSJ international conference on intelligent robots and system (IROS), pp 2661–2666
24. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE conference on computer vision and pattern recognition (CVPR), pp 580–587

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.