Enhancing Human-Robot Collaborative Object Search through Human Behavior Observation and Dialog

Takahiro Ishii Jun Miura Kotaro Hayashi
Toyohashi University of Technology
https://www.aisl.cs.tut.ac.jp/

Abstract

Human-robot collaborative object search entails joint efforts between a human and a robot operating in the same environment to locate a target object. Achieving efficient collaboration requires to avoiding duplicated search areas and sharing the space appropriately. This paper introduces a method for determining the robot's search strategy through the observation of human search behavior and engaging in dialog with the human. The behavior is determined by comparing estimated travel times for different behaviors with the actual elapsed time. When faced with multiple potential behaviors, the robot selectively generates informative queries to resolve ambiguities and obtain valuable responses. This occasional dialog activation serves as a crucial factor in achieving an efficient collaborative object search. Through collaborative experiments with real human subjects conducted in a virtual environment, we validate the effectiveness of our proposed method in reducing overlapped search areas and minimizing the time required to locate target objects.

1. Introduction

Mobile service robots have recently gained popularity for supporting people in various scenarios, such as delivery and attending. In the context of lifestyle support, finding and bringing a use-specified object is a common task for mobile service robots. It has been used as a typical task in many robotic competitions [14]. Another aspect of service robots is human-robot interaction. A human and a robot can collaborate physically or virtually to achieve a task. This paper deals with the problem where a human and a robot are in the same environment and collaborate to find a specific object.

One approach to collaboration is that the human takes the lead by giving commands or suggestions to the robot. There are systems with which humans can assist the robot's recognition or decision [12, 16, 13]. In this approach, a human has to know the robot's functionality to give appropriate commands and suggestions. Another approach is robotic support to human task execution [17, 6], where robots refer to the model of the task or the human state and decide the type and the timing of supportive actions. A robot and a human play an equivalent role in a collaborative object search task.

Collaborative object search can also be considered a distributed search problem. In the case of robot-robot collaboration, robots can easily share respective data through the network (e.g., [5]). In the collaborative search context, for example, one robot knows where the other robots have searched and are going to search and can choose its search area. However, in the case of human-robot collaboration, humans cannot give their knowledge digitally to the robot. Therefore, the robot must obtain it in other ways, such as observation and dialog, as in the case of human-human collaboration. In this paper, we take such an approach. The robot estimates the human's past search regions and predicts future search regions from observation. Moreover, it asks the human when further information is required for choosing the best robot action.

When humans collaborate for object search, they do not keep talking as usual conversation but usually conduct search independently and have a dialog only when information exchange is valuable. Therefore, we need to deal with the problem of what to ask [4, 15] and when to ask [20, 2]. This paper pursues these crucial problems in the human-robot collaborative object search context.

The contributions of this paper are threefold. First, we propose an approach to human-robot collaborative object search based on behavior observation and ambiguity-based dialog activation. Second, we developed methods to estimate the human's past search behavior and to predict future behavior when exploring an unknown environment. Third, we evaluate the proposed approach with real human subjects in a virtual environment.

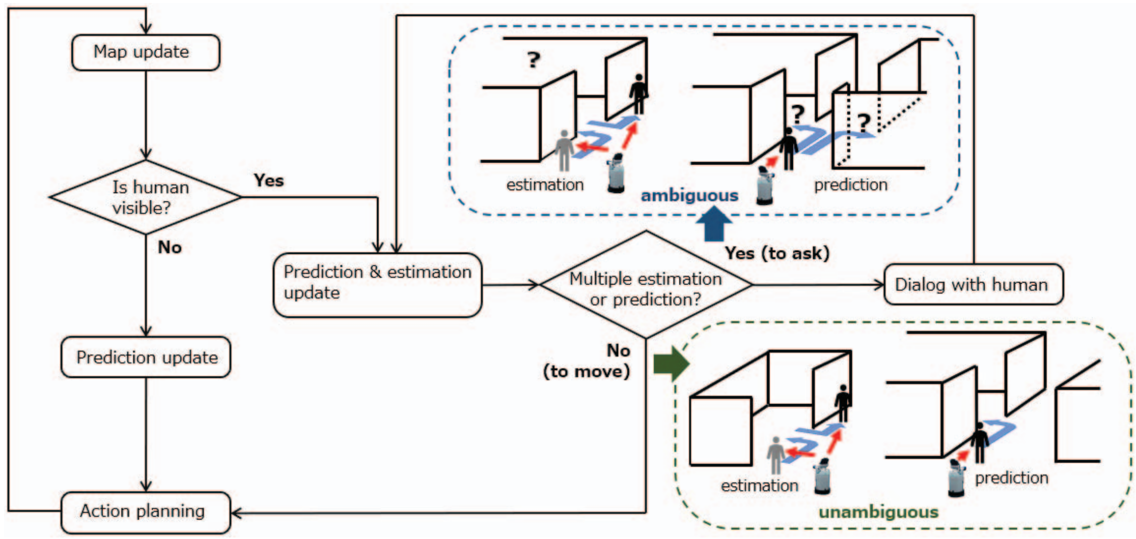


Figure 1. Diagram for robot's execution.

2. Related Work

2.1. Assistive robots

Assistive robots help humans in various contexts. One of the essential characteristics of such robots is the timeliness of assistance. Hanai et al. [9] developed a humanoid robot that can assist humans in pick-and-place tasks. The robot predicts human action to generate a collision-free robot hand motion timely. Hamabe et al. [8] developed a framework of programming by demonstration for collaborative assembly tasks. A learned model enables the robot to assist a human operator in a timely fashion. In these works, recognizing the intention of the human is crucial for effective collaboration. Xie et al. [23] proposed a method of estimating the intention of human search action and using the results for the robot to perform a complementary action. It is shown that such an implicit interaction reduces the human workload. In the case of collaborative object search tasks, as the robot often loses sight of the human, estimating the behavior while the human has not been observed is more critical.

2.2. Human assistance to robots

Sprute et al. [21] proposed an object search method with human assistance. A human operator monitors the robot and the environment using multiple smart cameras and suggests the target object location to the robot. Matsushita et al. [13] proposed a similar approach but with an omnidirectional camera interface on the robot. Burks et al. [1] proposed a framework to utilize human assistance, considering its uncertainty in planning an optimal robot action. These works deal with object search problems, but humans do not physically search in the same environment as the robot.

2.3. Dialog for human assistance

Knepper et al. [11] developed a framework for generating "targeted" requests in natural language, considering common contexts between the robot and the human. Morohashi and Miura [15] developed a method of generating queries for resolving the current ambiguity in the human's command. Deits et al. [4] developed a system that can interactively resolve ambiguities in user's commands, considering probabilistic relationships between symbols and recognized objects.

Cai and Mostofi [3] proposed a framework for optimizing a human-robot collaborative site surveillance problem, considering the robot's and the human's expected performance and the cost of observation and interaction. Serras et al. [19] developed a dialog system that uses the current information's entropy to decide if more information is needed. These works deal with "ask or not to ask" problems and can be used for human-robot collaboration with occasional dialog.

3. Outline of the Proposed Approach

3.1. Problem setting

We use the following problem settings. A robot and a human jointly search for target objects in the same environment. The environment is initially unknown, and the robot needs to map the environment using some SLAM method. All objects are put on tables. The number of the target objects is given, but their locations are unknown. The robot can detect and locate the human when he/she is visible. The search finishes when the human-robot team finds all target objects. Fig. 7 shows a scene of collaborative object search.

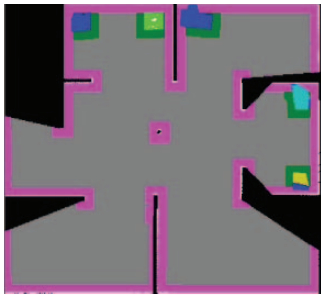


Figure 2. Labeled map example. Pink: occupied, gray: free, black: Unknown, blue: unsearched table, light green: table searched by robot, light blue: table searched by human, yellow: table to be searched by human. Dark green regions indicate margins for the robot to avoid collision with tables.

3.2. Execution Flow

Fig. 1 shows the flow of the proposed method. The robot manages the environmental information using a *labeled 2D grid map*. The robot updates the map at each frame using the latest sensor data. Then, the robot checks if the human is visible or not. If the human is visible, the robot estimates the regions where the human searched since the last time the human was visible. The robot also predicts the regions where the human will search for a fixed future duration. If the human is not visible, the robot extends the previous prediction by some fixed duration. To avoid a duplicated search of the same region by the human and the robot, if there are multiple possible estimated regions or predicted regions, the robot asks the human about his/her past or future behavior so that a unique estimation and prediction remain. Then, the robot updates the map labels and plans the best motion to search.

3.3. Mapping with table and object detection

The robot represents the environment using a 2D grid map with labels. We use GMapping [7] for occupancy grid mapping. In addition to three labels (free, occupied, unknown) commonly used in 2D occupancy grid maps, we use three more labels representing the table with three states: unsearched, searched by the human, to be searched by human, and searched by the robot. Tables are detected by extracting point data at a certain height. We use Yolo v3 [18] for object detection with a minimum distance requirement (currently, $3\,[m]$). Fig. 2 shows an example labeled map.

4. Estimation and Prediction of Human Behavior from Observation

An efficient collaborative search can be realized by appropriately sharing search areas between the robot and the human. To this end, the robot observes human behavior and estimates where the human has searched (estimation of human searched area in the past) and will search (prediction of the human search area in the future). The following subsections explain how to carry them out.

4.1. Estimation of human searched area

In the collaborative search, the robot and the human usually examine different areas. Therefore, the human is not always visible to the robot, and the robot intermittently sees the human in various places. One valuable information for the robot to determine its action is the areas where the human has searched during those unobserved periods. Our strategy to estimate such areas is to find a sequence of human visits on tables whose elapsed time is close to the time of an unobserved period. As objects are assumed to be on tables, we set an observation point for each table and suppose the human goes there to observe the tables. We first consider the known environment case and then extend it to unknown environments.

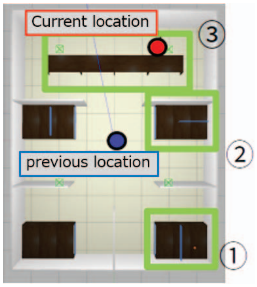
Known environment case First, we briefly describe a method of searched area estimation for the known environment case, that is, the case where we have a complete map of the environment with table positions. We search for sequences consisting of:

- the previous human position,
- observation points of the visited tables, *and*
- the current human position.

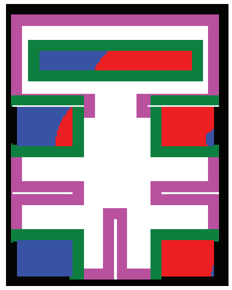
We do a breadth-first search with the previous human position as the root. The time of a route is the summation of the time of movements between nodes and that for observation at each table (set to $10\,[s]$), and we collect a set of routes whose estimated elapsed time is reasonably close to the duration of the latest unobserved period. This condition is used for pruning useless branches during the search. Once the most probable route is selected, the areas on the tables are marked as searched within a certain distance from the route.

The time for traversing between two locations is calculated using the Fast Marching Method (FMM) [22] on the grid map. By applying the FMM to a grid map with a single starting point (source point), we have a distance map, the pixel value of which indicates the distance from the starting point. The distance map is calculated for a table when the robot finds it for later use.

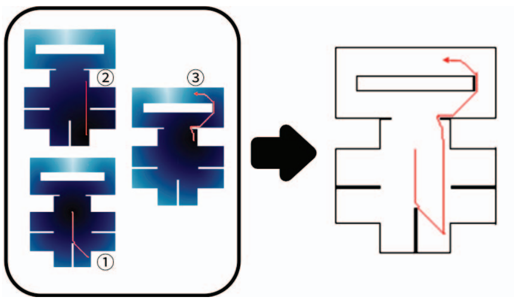
Fig. 3 shows an example of trajectory and searched area estimation. Fig. 3(a) shows a scene where the human is considered to visit three tables. Fig. 3(c) illustrates that using three FMM distance maps from the three tables, we can calculate the trajectory and the time of the whole sequence. We then have a map of searched areas shown in Fig. 3(b).



(a) Sequence of visited desks.



(b) Searched area estimation (red: searched region, blue: unsearched region).



(c) Trajectory estimation with FMM.

Figure 3. Example of searched area estimation.

Unknown environment case We then describe the method of estimating human-searched areas while exploring an initially-unknown environment. In this case, we need to consider unknown regions, as shown in Fig. 2. Therefore, we add another node type at *frontiers* [25] and allow each route to pass unknown regions.

A problem when a route includes an unknown region is that we cannot know how much time the human needs to traverse that region because no region size and table information is available. To cope with this problem, we set a minimum required time to traverse an unknown region (currently, $5\,[s]$) and allow any routes with unknown regions as long as this condition is satisfied. The tentative time to traverse an unknown region is calculated by subtracting the time for traversing known regions from the duration of the latest unobserved period. Also, once a previously-unknown region is observed and becomes known, the sub-route in that region is estimated as in the known environment case mentioned above.

Fig. 4 shows an illustrative example. On the left, there are two frontiers and two tables, and several candidate routes with one or two frontiers are generated, and the time for unknown regions is calculated. Suppose a route with two frontiers is selected, and the unknown region is observed later, as shown on the right. Then, the sub-route inside the observed region is estimated with the calculated time.

We also do a breadth-first search for candidate routes. Let T_{target} be the duration of the latest unobserved period and T_{known} is the time for traversing the part of a candidate route in the known region. Each candidate must satisfy either of the following:

- If a route is within only known regions (i.e., only table nodes in addition to the start and the goal node), $|T_{known} - T_{target}|/T_{target} < \theta_{diff}$;
- If a route includes frontier nodes, $T_{target} - T_{known} > \theta_{min}$,

where θ_{diff} is an allowed time difference against the target time, set to 0.2. θ_{min} is the minimum required time in an unknown region and is set to $5\,[s]$.

4.2. Prediction of human search area

Predicting where the human will search is also crucial for avoiding conflict in search areas between the human and the robot. We thus predict possible human motions (i.e., sequences of table and frontier nodes to visit) until a fixed future time point T_{future} for calculating the human's future search areas.

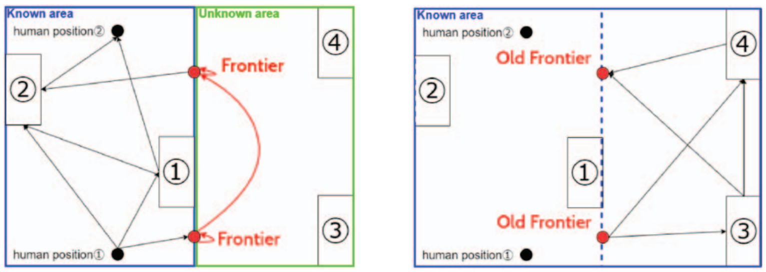
The only constraint we can use for prediction is the human position obtained by the latest observation. When there are many tables and frontiers, considering all possible combinations may cause a combinatorial explosion. We, therefore, take a best-first strategy with multiple choices for the first node to visit from the latest human position. We first calculate the time of movement to the nearest node, T_{min}. We then select other nodes whose movement time is less than $T_{min} + \theta_{max}$. We use each node as a first node and repeatedly choose the nearest unvisited nodes for route prediction until the total time exceeds T_{future}. We currently set T_{future} to $10\,[s]$ and θ_{max} to $3\,[s]$. Concerning frontier nodes, we assume the time for traverse an unknown region associated with a frontier is proportional to the size of the frontier.

When the human is not observed, we update the prediction with an extended time equivalent to the time elapsed from the last prediction. The prediction procedure is the same as above but uses the updated map information. If the first node in the last prediction is a frontier and no longer exists due to the map update, we use the nearest existing node as the first node to predict a similar motion to the one in the last prediction.

5. Robot Action Planning

5.1. To move or to ask

Choosing an action under a largely uncertain situation may result in inefficient consequences. In the case of collaborative object search, when there are many possibilities of



(a) Routes passing an unknown region. (b) Route estimation in the newly observed region.

Figure 4. Human behavior estimation in the presence of unknown regions.

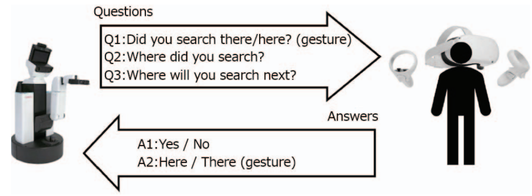


Figure 5. Dialog patterns.

regions where the human searched or will search, the robot may search the same place as the human. The robot should minimize the possibilities before deciding on the next action for an efficient collaboration. Therefore, we allow the robot to get more information from the dialog with the human.

In this paper, we take a simple, ambiguity-driven approach [15]; that is, if there are multiple possibilities for the human's past behavior or the human future behavior, the robot generates a query such that it can get a useful piece of information. If there is only one possibility for both the past and future behavior, the robot takes a search action explained below.

5.2. Dialog planning and execution

Various dialog patterns are possible in actual collaboration situations. As recognition of speech or text is out of the scope of the current work, we prepare a set of Q&A patterns shown in Fig. 5. Combined with a pointing gesture by the robot or the human, the robot can acquire the necessary information on the human's past and future behavior.

Query generation We use the following simple rules for query generation:

- If there are multiple past routes, sort the routes with the closeness to the target time T_{target}, and let ΔT_{r1} and ΔT_{r2} be the time difference of the best route $r1$ and the second-best route $r2$.
  - If $\Delta T_2 - \Delta T_1 > \theta_{diff}$, ask **Q1**:"Did you search there (here)?" with pointing the route r_1.
  - Otherwise (i.e., there is no distinctive route), ask **Q2**: "Where did you search?"
- If there are multiple future routes, ask **Q3**: "Where will you search next?"

To start a dialog, the robot tells the human that it has questions and comes near to some distance (currently, within $4\,[m]$).

Processing human response We use the following procedure to process the human response to the question:

- If the question is **Q1**, the answer is **A1**:
  - If the answer is "Yes", the robot confirms that the human took the route used in the question.
  - If the answer is "No", the robot excludes that route from the candidate set.
- If the question is **Q2**, the human's response is **A2**: "Here (there)" with a pointing gesture. The robot calculates the pointed location from the human position and the pointing direction and chooses one route closest to the pointed location.
- If the question is **Q3**, the human's response and its processing are the same as in the previous case.

5.3. Search action planning

The robot takes one of the following actions:

- Object search on the table when there is a table with unsearched areas.
- Frontier-based exploration when all known tables are searched.

For the first type of action, the robot takes the next best view (NBV) approach. We place a set of viewpoints around the table under consideration, and choose the best one which maximizes the following score function: $score = S_c/(T_{mot} + T_{obs})$, where S_c is the size of the currently-unknown area on the table to be observed by a viewpoint, T_{mot} is the time cost to reach the viewpoint from the current location, and T_{obs} is the time cost for observation there. This score evaluates the reduction of unknown areas per time. For the second type of action, we choose the nearest frontier as the current destination.

For an efficient sharing of search space, the robot needs to consider not only its search history but the human's past and future search behavior. To this end, we record those types of information on the labeled map; the robot excludes

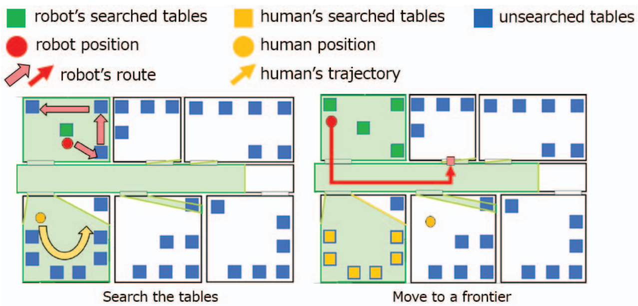


Figure 6. Example action planning scenarios. (left) search tables; (right) move to a frontier.

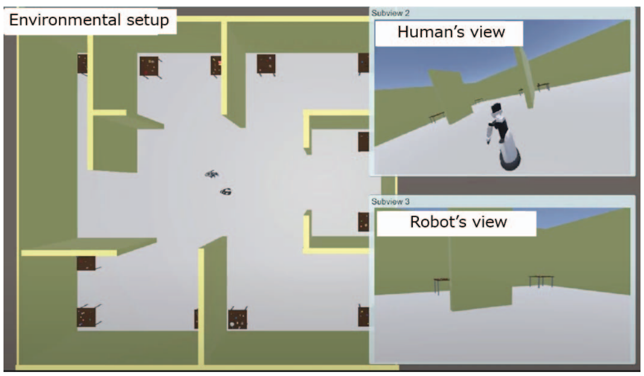


Figure 7. SIGVerse simulation environment. (left) Environment setup. (top-right) human avatar's view. (bottom-right) robot's view.

the regions the human searched or will search from the candidate target of its own. Fig. 6 shows example scenarios of robot action planning.

6. Experiments

6.1. Simulation environment

We implemented the experimental environment using SIGVerse [10] environment. In SIGVerse, we can enter a virtual environment as an avatar and interact with robots and other avatars. We use a model of Toyota's HSR (Human Support Robot) [24] as the simulated robot and control it from the ROS environment. Fig. 7 shows the simulated environment.

To enter the SIGVerse environment, we utilize a Meta Quest 2 VR headset. Using the headset, we employ the following method to interpret human responses. First, we allocate two different buttons for answering "yesr" and "no" respectively. Additionally, we assign another button for responses involving pointing gestures such as "here" or "there." This response is requested when the robot is faced with multiple candidates of the human's past or future behavior, and it needs to select one. Using the provided pointing gesture, we calculate the direction of the pointing with respect to the current avatar location, and choose the best candidate whose direction aligns closest to the indicated pointing direction. Fig. 8 shows a scene where the human avatar responses with the pointing gesture.

Figure 8. Response with pointing gesture.

6.2. Experimental procedure

We put three target objects in the environment. There are three patterns of target object placement, and one of them is randomly selected for each trial. We tell a subject to search the environment for three target objects with the robot, and the robot occasionally asks questions. We also teach the subject how questions are presented and how to respond to them using a controller device.

We compare the following three methods:

(1) The method that does not consider human behaviors (baseline).

(2) The method that carries out both human behavior estimation and prediction. In estimation, all of the possible routes are considered. In prediction, only the best one is considered.

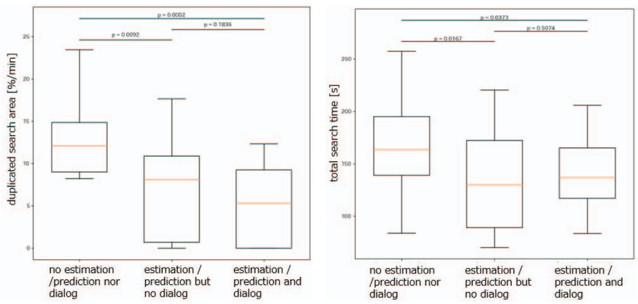
(3) The method that carries out human behavior estimation and prediction and performs dialog when necessary (proposed).

Eleven subjects (male: 9, female: 2) participated in the experiments. Each subject tries each method twice, six times in total. We compared duplicated search areas by the robot and the human and total search times among the methods.

6.3. Results

Fig. 9 shows the comparison of the robot and the human trajectories for the three methods. Fig. 10 shows example trajectories in one of the experiments in each condition.

The results show that the estimation and the prediction of the human behavior can significantly reduce the duplicated search area and the search time. Without such estimation and prediction, the robot and the human search the same region, as shown in the top-left of Fig. 10.



(a) Comparison in duplicated search. (b) Comparison of total search time.

Figure 9. Comparison of the three methods.

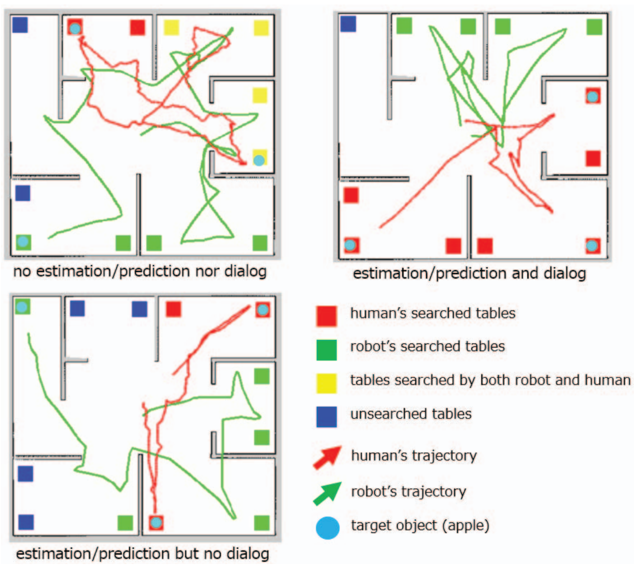


Figure 10. Example trajectories in each method. (top-left) no estimation/prediction nor dialog; (bottom-left) estimation/prediction without dialog; (top-right) estimation/prediction and dialog.

Concerning the effectiveness of dialog, we cannot see statistically significant differences. The dialog can slightly reduce the duplicated search area but increases the total search time. One possible reason is that in the current implementation of the dialog function, the robot needs to come near the human to start a dialog, which requires extra robot movements. In the experiments, the robot moved for the dialog for $15.6\,[s]$ on average, which is large enough considering the robot's speed and the size of the environment. In human-human collaboration, they can talk from a distant place, but recognition of pointing gestures may be degraded. We need to elaborate dialog-based interaction to be more realistic. Another possible reason is that the reduction of search region is not significant as expected. There was a case where the human searched the region the robot had previously searched. The current dialog is for the robot to obtain missing information from the human; the human cannot ask the robot but gets such information only from observation. Extending the dialog to full two-way (i.e., the human can also ask the robot) will improve the efficiency of the collaborative search. Regarding the variance of results, the method with dialog seems to have an advantage. This result is probably because the method without dialog does not wait until ambiguities are resolved, and the robot may happen to take an inefficient search action.

7. Summary

This paper deals with collaborative object search as a typical problem for human-robot interaction research. We have implemented a method to estimate and predict human past and future behavior. To cope with multiple estimation and prediction situations, we introduce a dialog capability by which the robot can obtain helpful information from a human. We implemented an experimental system using a virtual environment SIGVerse and conducted evaluation experiments with human subjects. The results show that the estimation and the prediction of human behavior significantly improve the efficiency of the collaborative search. Introducing the dialog capability seems effective in reducing the stability of the performance, but various extensions will be necessary for further improving the efficiency.

References

[1] L. Burks, N. Ahmed, I. Loefgren, L. Barbier, J. Muesing, J. McGinley, and S. Vunnam. Collaborative human-autonomy semantic sensing through structured pomdp planning. *Robotics and Autonomous Systems*, 140, 2021. 2

[2] H. Cai and Y. Mostofi. To ask or not to ask: A foundation for the optimization of human-robot collaborations. In *Proceedings of 2015 American Control Conf.*, pages 440–446, 2015. 1

[3] H. Cai and Y. Mostofi. Human-robot collaborative site inspection under resource constraints. *IEEE Trans. on Robotics*, 35, 2019. 2

[4] R. Deits, S. Tellex, P. Thaker, D. Simeonov, T. Kollar, and N. Roy. Clarifying commands with information-theoretic human-robot dialog. *J. of Human-Robot Interaction*, 1(1):78–95, 2012. 1, 2

[5] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart. Distributed multi-robot exploration and mapping. *Proceedings of IEEE*, 94(7):1325–1339, 2006. 1

[6] H. Goto, J. Miura, and J. Sugiyama. Human-robot collaborative assembly by on-line human action recognition based on an fsm task model. In *Proceedings of HRI2013 Workshop on Collaborative Manipulation: New Challenges for Robotics and HRI*, 2013. 1

[7] G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Trans. on Robotics*, 23:34–46, 2007. 3

[8] T. Hamabe, H. Goto, and J. Miura. A programming by demonstration system for human-robot collaborative assembly tasks. In *Proceedings of 2015 IEEE Int. Conf. on Robotics and Biomimetics*, pages 1195–1201, 2015. 2

[9] R. Hanai, R. Oya, T. Izawa, and M. Inaba. Motion generation for human-robot collaborative pick and place based on non-obstruction strategy. In *Proceedings of 2011 IEEE Int. Conf. on Robotics and Biomimetics*, pages 20–25, 2011. 2

[10] T. Inamura and Y. Mizuchi. Sigverse: A cloud-based vr platform for research on multimodal human-robot interaction. *Frontiers in Robotics and AI: Human-Robot Interaction Section*, 2021. 6

[11] R.A. Knepper, S. Tellex, A. Li, N. Roy, and D. Rus. Recovering from failure by asking for help. *Autonomous Robots*, 39:347–362, 2015. 2

[12] M. Mast, M. Burumester, B. Graf, F. Wiesshardt, G. Arbeiter, M. Španěl, Z. Materna, P. Smrž, and G. Kronreif. Design of the human-robot interaction for a semi-autonomous service robot to assist elderly people. In *Ambient Assisted Living*, pages 15–29. 2015. 1

[13] M. Matsushita, K. Hayashi, and J. Miura. An operator and obserer style teleoperation user interface for remote object search. In *Proceedings of 2021 IEEE/SICE Int. Symp. on System Integration*, 2021. 1, 2

[14] R. Memmesheimer, V. Seib, and D. Paulus. Winning team of the robocup@home open platform league 2017. In *RoboCup 2017: Robot World Cup XXI*, pages 509–520. Springer International Publishing, 2018. 1

[15] K. Morohashi and J. Miura. Query generation for resolving ambiguity in user's command for a mobile service robot. In *Proceedings of 2019 European Conf. on Mobile Robots*, 2019. 1, 2, 5

[16] K. Nagahama, K. Takeshita, H. Higuchi, K. Yamazaki, T. Yamamoto, and M. Inaba. Estimating door shape and manipulation model for daily assistive robots based on the integration of visual and touch information. In *Proceedings of 2018 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 7660–7666, 2018. 1

[17] S. Nikolaidis and J. Shah. Human-robot teaming using shared mental models. In *Proceedings of HRI2012 Workshop on Human-Agent-Robot Teamwork*, 2012. 1

[18] J. Redmon and A. Farhadi. YOLOv3: an incremental improvement. arXiv:1804.02767, 2018. 3

[19] M. Serras, N. Perez, M.I. Torres, and A. del Pozo. *Entropy-Driven Dialog for Topic Classification: Detecting and Tackling Uncertainty*, pages 171–182. Springer Singapore, 2017. 2

[20] T. Spexard, S. Li, B. Wrede, M. Hanheide, E.A. Top, and H. Hüttenrauch. Interaction awareness for joint environment exploration. In *Proceedings of 16th IEEE Int. Symp. on Robot and Human Interactive Communication*, 2007. 1

[21] D. Sprute, A. Pörtner, R. Rasch, S. Battermann, and M. König. Ambient assisted robot object search. In *Proceedings of 2017 Int. Conf. on Smart Home and Health Telematics*, pages 112–123, 2017. 2

[22] A. Valero-Gomez, J.V. Gomez, S. Garrido, and L. Moreno. Fast marching methods in path planning. *IEEE Robotics and Automation Magazine*, 20(4):111–120, 2013. 3

[23] D. Xie, Y. Lin, R. Grupen, and A. Hanson. Intention-based coordination and interface design for human-robot cooperative search. In *Proceedings of 2011 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2011. 2

[24] T. Yamamoto, K. Terada, A. Ochiai, F. Saito, Y. Asahara, and K. Murase. Development of human support robot as the research platform of a domestic mobile manipulator. *ROBOMECH journal*, 2019. 6

[25] B. Yamauchi. A frontier-based approach for autonomous navigation. In *Proceedings of the 1997 IEEE Int. Conf. on Computational Intelligence in Robotics and Automation*, pages 146–151, 1997. 4