

Intention-Based Coordination and Interface Design for Human-Robot Cooperative Search

Dan Xie, Yun Lin, Roderic Grupen and Allen Hanson

Abstract—In this paper, a multi-agent search scheme is presented that supports the recognition of activities and, thus, learning methods for cooperative human-robot interaction. In our approach, stochastic models of human search activity are used to estimate state for HRI. The robot updates a Probabilistic Distribution Function of the target object using the observations and the estimated state of human peers. By this means the robot can choose places to search to compensate the behavior of human peers. This paper also presents an implicit interface design for robot assisted tasks, which allows the robot to infer the intention of the user and to provide assistance autonomously. It reduces the cognitive workload of the user and therefore is useful for elder care applications. The effectiveness and the efficiency of the proposed approaches are demonstrated in the experimental results.

I. INTRODUCTION

There has been considerable recent interest in addressing the problem of robot search. The applications range from emergency response to in-home elder care. In search operations, a team of intelligent agents can provide a robust solution with greater efficiency than can be achieved by single agents, even with comparatively superior mobility and sensors. Our vision for search problem solving is that human and robots will work as partners, leveraging the capabilities of each. Human-robot teams are used in Urban Search and Rescue (USAR) [9], [10]. In these work robots are teleoperated and used mainly as tools to search for survivors and objects. There is inadequate computational support for coordinating human and robots as peers in search tasks, which is the main focus of this paper. In our approach the robot learns to recognize and complement a human’s search activity.

Recently intension estimation for robot human interaction has been researched to make robot assist human. There has been recent work aimed at recognizing human activity as a means of inferring intention. In [13], a vision based approach is used to infer the intentions of other agents. In [11], a hierarchical hidden Markov model is used to recognize a set of complex indoor activities. However, the approach of integrating intention estimation with robot planning is still in need of investigation, especially in search tasks.

In this paper we present a multi-agent cooperation scheme for search tasks. The autonomous search agent maintains estimates of the probability density function (PDF) for the object location and makes independent decisions about its

search process. Cooperation is achieved by sharing perceptual information and intention among cooperating agents. For robot teammates, explicit message transmission is used. The human teammate is modeled as an agent without this mode of communication. In this case, the robot agents infer the current state and intention of the human peer using a model of human search activity acquired in the learning session. By inferring human search states, the robot chooses compensative actions to achieve efficient cooperation with human peers.

A second area of our work is the problem of user interface design for robot assistance. Cognitive load is an important factor for human robot interaction and has been studied considerably in the work on interface design [7], [6], [8]. Most of this work is based on explicit message transmission without considering the potential for using predictions of human intention. To reduce the mental stress in H-R interaction, collaborative control [3], [5] was developed for mobile autonomous robots. The robots work autonomously until they run into a problem they can’t solve. At this point, the robots ask the remote operator for assistance, allowing human-robot interaction and autonomy to vary as needed. In this paper, we discuss how collaborative control mechanisms can be used for service robots in elder care applications. An implicit interface design for robot assisted tasks is proposed, which allows the robot to infer the intention of the user and to provide assistance autonomously. It reduces the cognitive workload of the user and therefore is useful for elder care applications.

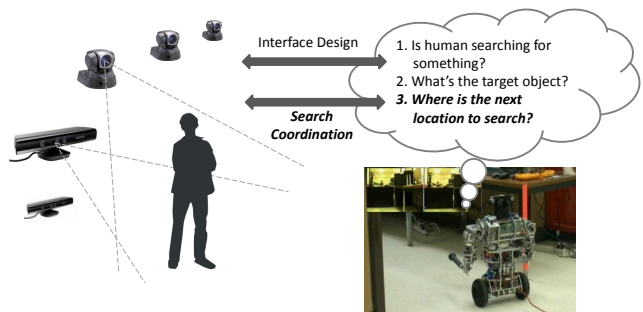


Fig. 1. A scenario of robot assisted search in home environment

The robot assisted search scenario is illustrated in Fig. 1 where some of the information that the robot needs to know for efficient assistance is listed. The first two questions are related to user interface design, which is used to inform the robot what assistance is needed by human. The third question is related to search coordination, where the robot

This work is funded by ONR/MURI award N00014-07-1-0749.

The authors are with the Department of Computer Science, University of Massachusetts Amherst. {dxie, yunlin, grupen, hanson}@cs.umass.edu

needs to plan the next move to cooperate with its human peer in order to achieve the goal. In this paper we will present our approach to search coordination first, and then discuss the problem of user interface design.

This paper is organized as follows. Section 2 presents our approach of human-robot cooperative search. An implicit user interface design is proposed in Section 3. In Section 4 we present the experimental setup and results. Conclusions and future work are presented in Section 5.

II. HUMAN-ROBOT COOPERATIVE SEARCH

A. Bayesian Search Strategy

The search problem can be represented in a Bayesian framework, in which for a target r , the state vector of its location $\vec{x}_r \in \mathcal{X}_r$ in \mathcal{S} is expressed in the form of a probability distribution function (PDF) $p_r(\vec{x})$. Given a prior PDF $p_r(\vec{x}_0|z_0) \equiv p_r(\vec{x}_0)$ of the target and the independent observations z , the PDF at time step t can be constructed recursively using Bayes' theorem. In the application of object search to a living space, it is reasonable to assume that when the search process starts, the target object is stationary and not allowed to move until the search finishes. So we have $p_r(\vec{x}_t|z_{1:t-1}) = p_r(\vec{x}_{t-1}|z_{1:t-1})$. After each observation, the PDF is updated according to the observation,

$$p_r(\vec{x}_t|z_{1:t}) = K p_r(\vec{x}_{t-1}|z_{1:t-1}) \cdot p_r(z_t|\vec{x}_t) \quad (1)$$

Where K is the normalization factor and is given by,

$$K = 1 / \int [p_r(\vec{x}_t|z_{1:t-1}) p_r(z_t|\vec{x}_t)] d\vec{x}_t \quad (2)$$

A “one-step-lookahead” strategy [14] that maximizes $p(Z_t|z_{1:t-1})$, where Z_t represents a “detection” event at time $t - 1$, is used with very low computational overhead.

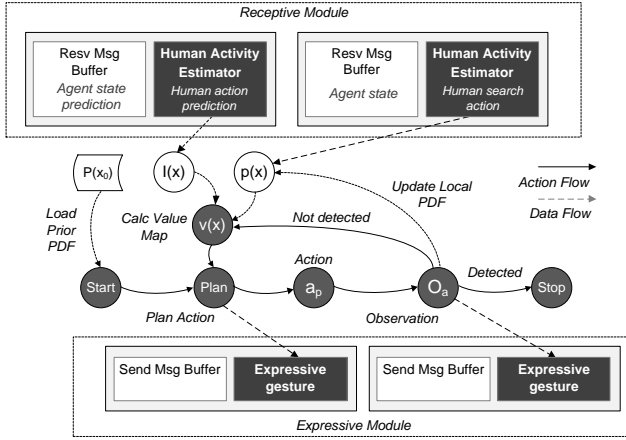


Fig. 2. Cooperative search strategy for a single agent

Fig. 2 shows a schema-based representation of the search behavior for a robot agent. The agent shares information with other teammates using a Receptive module that receives the perceptual information and intention of other agents, and an Expressive module that broadcasts its observation result and planned intention. In our previous work [14] explicit

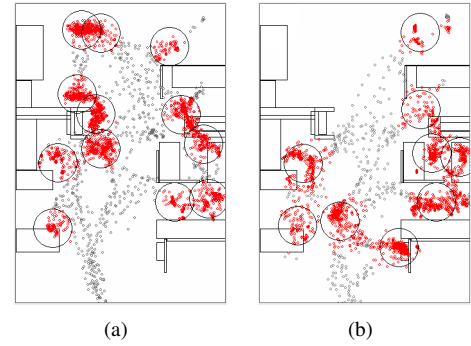


Fig. 3. Human data (a) Searching for a book; (b) Searching for a screwdriver.

message transmission was used to coordinate autonomous searching agents, where the *Resv Msg Buffer* and *Send Msg Buffer* are used. An agent updates its local PDF map and plans its actions accordingly when receiving observations and the “next move” intentions from the other agents. For more details of intelligent agent cooperative search using this message sharing method refer to [14].

B. Human Peer Modeling

Human peers can be modeled as agents without explicit message transmission. As shown in Fig. 2, the *Human Activity Estimator* and *Expressive Gesture* are used to replace the message sharing modules to communicate with human peers. In this paper, we focus on the receptive module that enables the robot to estimate human activity and to provide complementary behavior. At time t the human activity estimator should provide two types of information: (i) Human action sequence $X_t = (x_0, x_1, \dots, x_t)$, which is the sequence of locations human has been in a search trial; and (ii) Human action prediction x_{t+1} , which is the predicted location of human in time $t + 1$.

1) *Dwelling State Detection*: A reasonable human observation model is necessary for the robot to estimate the impact of human search actions. Our experimental data reveals that a human searcher will dwell in places where objects are likely to be found, which suggests using human observation models with different kernel sizes for dwelling and walking states. Therefore, a preprocessing step is used to distinguish dwelling and walking states for all human tracking points $o(t) = [x, y, d_x/d_t, d_y/d_t]$. A K-means algorithm is used to cluster the state vectors and to distinguish if the tracking points are in dwelling or walking states. For cluster C_i , if $\sqrt{\bar{d}_{x_i}^2 + \bar{d}_{y_i}^2} > T_{walk}$ (T_{walk} is a predefined threshold), all state vectors $o_k^i \in C_i$ are considered as walking states.

2) *Activity Modeling with HMM*: Given observed state vectors, stochastic model can be constructed for human searching activities. HMMs are powerful tool for modeling sequential phenomena, and have been successfully used in applications involving speech and sound. Recently, HMMs have been used for activity understanding, showing a significant potential for their use in activity modeling and inferring intent [12].

In our work HMMs are used to model the search activity of human beings, where the hidden states denote search locations, and the observed outputs are human tracking results with the camera network. HMM parameters $\lambda = (A, B, \pi)$ are trained by observing the action sequence of the human subject for a period of time. the Baum-Welch algorithm [12] is used to train model parameter λ_i corresponding to the search activity class c_i . Equation 3 modifies transition weights and statistics of the models using a generalized expectation-maximization algorithm.

$$P(O|\lambda) = \sum_X P(O, X|\lambda) \quad (3)$$

Given a trained HMM with parameters $\lambda = (A, B, \pi)$, the Viterbi algorithm (Equation 4) produces the most probable successive states from an observation sequence.

$$P^*(O|\lambda) = \arg \max_{\{X\}} P(O, X|\lambda) \quad (4)$$

where the joint distribution can be factorized as follows,

$$P^*(O, X|\lambda) = p(x_1|x_0) \prod_{t=1}^T p(x_t|x_{t-1})p(o_t|x_t) \quad (5)$$

Human action sequence $X_t = (x_0, x_1, \dots, x_t)$ represents the locations where human subjects paused in the course of a search trial. Human observation model $p_h(z_t|x_t)$ can be represented by a Gaussian distribution for simplicity. Using X_t and p_h , the robot updates the PDF using Bayes' theorem (Equation 1). Given the current hidden state x_t and the transition matrix A , the state x_{t+1} can be predicted, which indicates the location where a human subject is likely to make the next observation. The robot selects actions based on the inferred hidden state in the following way:

(1) At time t , the robot infers the hidden action sequence X_t and predicts the next human search location x_{t+1} .

(2) The robot selects the action a_t to complement the search behavior of the human. Applying Bayes' theorem, a temporary PDF P'_t is constructed by updating robot's PDF P_t with the estimated human location sequence X_t along with the human observation model p_h therein. x_{t+1} is used as an Inhibition Message $I_t(x)$ [14] to update P'_t as well to bias the robot to avoid searching locations where the human subject is likely to go in the next step.

(3) The robot uses P'_t to plan the next action. A value v_i is calculated for each grid node indicating the benefit of visiting this node. Given P'_t for an agent, v_i is calculated by,

$$v_i = \sum_{g_k \in V_i} p'_t(x_k) \quad (6)$$

where V_i is the set of all grid nodes in the observation field, i.e., that can be observed when the agent is visiting grid node g_i .

(4) The robot moves to the planned location and perform observation. The PDF P_t of robot is updated in Bayes' theorem using its observation result. Fig. 4 illustrates the

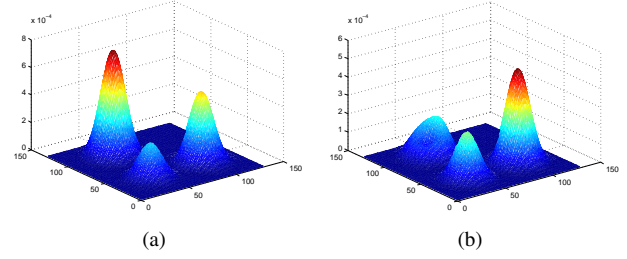


Fig. 4. PDF update for a search agent. (a) Before observation. (b) Update agent's PDF with current observation.

update of PDF map using the Bayes' theorem. The steps (1)-(4) iterate until the target object is found by the robot or human.

3) *Learning from Demonstration*: This paper adopts the perspective that cooperative multi-agent systems depend on independent skills and behavior in individual agents. We previously proposed using the *detection history* and *daily activity density* of the user to refine the prior PDF over objects to improve search efficiency [14]. In this paper we propose to refine the prior PDF over objects using *search activity density (SAD)* of human. SAD reflects the spatial probability of an object by observing the behavior of human demonstrations of search activity directed at the object.

The influence of a dwell point demonstrated by a human demonstrator on the prior PDF is determined by the Cartesian observation error ellipsoid, which can be estimated by the triangulation Jacobian J for a camera pair. If D is the baseline between two cameras and γ_R and γ_L are the respective headings to the target, the uncertainty Jacobian is given as follows,

$$J = \frac{D}{\sin^2(\gamma_R - \gamma_L)} \begin{bmatrix} \sin \gamma_R \cos \gamma_R & -\sin \gamma_L \cos \gamma_L \\ \sin^2(\gamma_R) & -\sin^2(\gamma_L) \end{bmatrix}$$

The eigenvalues and eigenvectors of JJ^T define the principle directions of error amplification in stereo triangulation. The probability of observing target r at an occupancy grid \vec{x} is given by:

$$p_h(\vec{x}|r) = N_h \sum_{k=1}^{D_i} K_h(\vec{x} - \vec{x}_k) \quad (7)$$

where \vec{x}_k represents the locations where human dwell states are observed, D_i is the total number of dwell points, and $K_h(\cdot)$ is a suitable kernel function (here, a Gaussian), which is scaled and rotated using the eigenvalues and eigenvectors of JJ^T .

The robot should be able to compensate for the actions of a human instead of simply imitating the behavior. It should use a combination of *detection history*, *daily activity density* and *searching activity density* to build a prior PDF. In addition, from the demonstration of human search, it is possible to infer where human do not search (e.g., the area underneath the table is inaccessible to many elders). When the robot is updating the prior using the *detection history*, the location x

has higher weight to affect the prior PDF if $x \in \mathcal{A}$, where \mathcal{A} is a set of locations that are inaccessible to the human. The robot learns to examine the places where the human does not go, and to compensate for search preferences demonstrated by human beings.

III. INTERFACE DESIGN FOR IMPLICIT COOPERATION

To cooperate with human peers autonomously, the robot also needs to know when to search and what is the target object (first two questions in Fig. 1). This information is usually conveyed through a GUI or voice control and may increase the cognitive load of the user. This problem becomes more severe in elder care.

In collaborative control frameworks, robots work autonomously until they run into a problem they can't solve. At this point, the robots ask a remote operator for assistance. A similar mechanism is used in our work, where the robot estimates the human's intention and autonomously provides assistance without explicit command instructions to do so. We call this approach as Implicit Interface for service robot. The explicit and implicit user interfaces are categorized as follows:

- (i) an explicit interface conveys explicit tasks to the robot to implement a coordination plan conceived by the operator.
- (ii) an implicit interface infers robot tasks from the observations of human activity. The robot asks questions to verify the inference results when the recognition confidence is low. The autonomy of the robot is adjusted dynamically according to the recognition confidence and the cognitive load of the user.

A Hidden Markov Model (HMM) is used in this section to recognize human activities. It is possible to train the recognition on a partial sequence of the human search locations and then perform a classification on the observation data. Our experiments show that search activities for different objects are distinguishable since human search patterns different for different objects.

Given a trained HMM $\lambda = \{A, B, \pi\}$, the probability of an observation $O = \{o_1, o_2, \dots, o_n\}$ can be calculated. The model with maximum likelihood is chosen as the recognized class. The activity classification problem can be formulated to identify the class $c_i, (i = 1 : N)$ to which the trajectory state sequence belongs. The basic formulation of the problem is given by the maximization of a conditional probability as in Equation 8. The classes are considered to be balanced in our experiments.

$$i^* = \operatorname{argmax}_x P(c_i|O) = \operatorname{argmax}_i \frac{P(O|c_i)P(c_i)}{P(O)} \quad (8)$$

The processes that support the implicit interface are elaborated as follows.

- (a) The robot waits and observes human activities. After an amount of data accumulated, the classification algorithm is used to classify the observed human activities.
- (b) If activity a_i is detected (e.g., human is searching for a book), the robot can choose to initiate a dialog with the

human for verification. The cognitive load M_t of user at time t is to be limited to satisfy a threshold T_{ml} throughout the interaction. The robot initiates a dialog if $M_t < T_{ml}$. Otherwise, the dialog is not allowed and the robot continues to assist autonomously until better recognition results obtained. Cognitive load can be described in several different ways. For simplicity, M_t is described in terms of verbal communication density, which is the weighted sum of questions in a period of time.

- (c) Given user feedback on the verifying activity a_i , the robot starts assisting human.

In step (b), more information can be obtained from the dialog with the human, such as which objects are related to the activity a_i . When generalizing the implicit interface from search activity to other daily activities, it is important to infer the objects that participate in a_i . For search activity, usually only one object involved. For other activities like reading, multiple objects may be involved. Imagine an example of applying the implicit interface to a reading activity. Two objects (a book and a light) can be associated to this activity in the form of prior knowledge. Due to the constraints on cognitive load, the robot may only be allowed to assist autonomously, in which case, robot will bring the book to the user and turn on the light even it is not verified through the dialog.

IV. EXPERIMENTAL RESULTS

A. Experimental Setup

Experiments in this paper were performed in a mock apartment, which is 42×28 square foot a lab environment as shown in Fig. 5 (a).

A hybrid camera network is deployed in our test environment for human tracking. The camera network consists of four color cameras (Sony EVI-D100 PTZ cameras) and two ranging cameras (Microsoft Kinect sensors). The color cameras perform tracking on captured frames using color and edge features. Each time step, a pair of color cameras is selected to determine the 3D location of people in the environment. The ranging camera uses first order recursive filter: $B_m = (1-\alpha)B_m + I_c$ to integrate the new depth image I_c into the current 3D background model B_m , and tracks the human in 3D space. The tracking results from color cameras and ranging cameras are merged to provide global tracking.

The uBot-5 [4] is used as the mobile robot. A stereo vision system with pan-tilt platform is mounted on the robot to perform visual search. In this paper, since we focus on investigating the receptive behavior and there are no expressive gestures performed by the robot, it is reasonable to assume that human behavior is not influenced by robot actions. The real world data of the human peer is recorded and replayed in simulations, in which the simulated uBot-5 cooperates in search tasks to complement the human teammate. The simulated uBot-5 and environment were developed using the Microsoft Robotics Development Studio. The target object is simulated as a colored 3D model that can be visually detected by the simulated uBot-5 using Camshift [1] algorithm. In addition to the simulated environment, our group also has

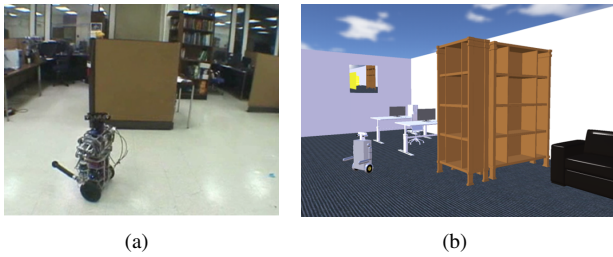


Fig. 5. (a) Physical robot (b) Simulated robot and environment for experiments

began to experiment with real robots that interact with and without human search activities (Fig.5). We posted a video showing the qualitative result of H-R cooperative search in a real lab environment at [2].

B. Efficiency of Human-Robot Cooperative Search

We first evaluate the efficiency of the proposed cooperative search strategy. Eight subjects of convenience were recruited for this study. Among these participants, 3 were colleagues of the authors and are familiar with the lab environment. The remaining 5 were unfamiliar with the search room and received a short description of the room configuration and furniture. Two search activities were considered, *searching for a book* and *searching for a screwdriver*. In each trial the target object was placed randomly in possible locations (e.g., a book on the shelf, table or on the couch) in the environment. The participants are asked to search for a designated target objects in each trial. As shown in Fig. 3, subjects presented different search patterns for different objects. For instance, the participants go to bookcases and tables to search the book, and toolboxes when searching for the screwdriver. Each participant performed 5 search trials for each object for a total dataset of 40 trials. 24 trials were used to train the activity model and the remaining 16 trials were used to evaluate the result.

We compare the efficiency of four search strategies: (1) Single human search; (2) Single robot search; (3) H-R team search without cooperation, where human and robot search simultaneously but independently without communication; (4) H-R team search with cooperation, which is the proposed strategy. For the 16 datasets used for evaluation (for each object), the “time to detect” cost of different search strategies were measured. The experiments in this part use the explicit interface, where subjects send command to robot directly, and the team starts searching together.

Table I gives the average time cost of the search strategies. It can be seen that the search efficiency with the single robot is comparable to human (The average time cost to find an object on all trials is 211.9s for single human and 219.8s for single robot), which indicates that the robot is a qualified search teammate for efficient cooperation. The time cost of cooperative search is 29.6% less in average than that of non-cooperative search for object *book*, and 13.4% for object *screwdriver*, which indicates that the proposed cooperative search is efficient.

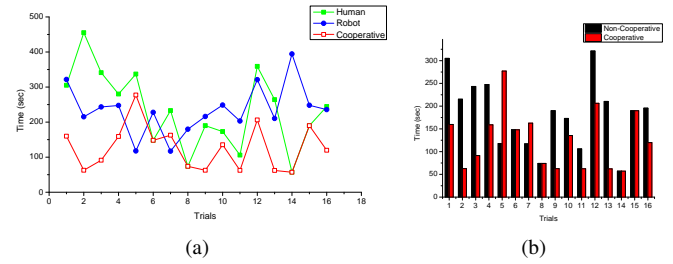


Fig. 6. Search efficiency (for Book)

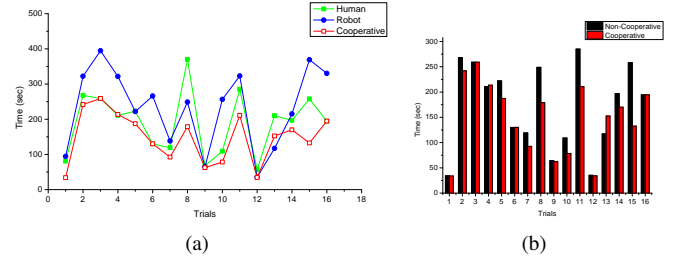


Fig. 7. Search efficiency (for Screwdriver)

Fig. 6 shows the comparison of search strategies in all trials for searching the book. In all trials the cooperative search is better than single human search. In 14 out of 16 trials the proposed cooperative search strategy is more efficient than the single robot search without human peer. In 14 out of 16 trials the performance of the proposed cooperative search is better than or equal to that of the non-cooperative strategy. There are some cases where cooperative search is worse than non-cooperative strategy. The reason is that sometimes the human will miss the target object when checking a place, therefore inhibits the robot from going to the same place and finding it. It suggests an improvement of our approach by learning different observation models for the human teammates with different searching capabilities, which is beyond the scope of this paper. Fig. 7 shows similar results for searching for the screwdriver. The experiments show that the proposed cooperative search is efficient.

TABLE I

AVERAGE TIME COST WITH DIFFERENT SEARCH STRATEGIES. (SEC)

	Human	Robot	No-Co	Cooperation
Book	234.1	217.6	181.0	127.4
Screwdriver	189.7	221.9	170.5	147.2
All Data	211.9	219.8	175.8	137.3

C. Interface Design Experiments

In this part the search efficiency of the explicit and implicit user interface is compared. Firstly we measure the classification accuracy of the partly observed trajectory of subjects when searching for the book and the screwdriver. Fig. 8 shows that the accuracy improves with the number of observations. Given 40 seconds of observation, the robot is able to predict the search activities with 75% accuracy. It can be used as the time for the robot to initiate the dialog with

human to clarify the remaining ambiguity, if the subject's cognitive load is lower than a threshold $M < T_{ML}$. In the case of $M > T_{ML}$, the robot waits longer to collect more data for better human activity recognition.

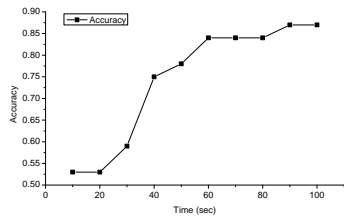


Fig. 8. Accuracy in search activity prediction

Then we evaluate the search efficiency when using implicit interface design. With implicit interaction, the robot needs time to predict what human is looking for, which causes a delay for the subsequent cooperative search. We want to evaluate if this delay causes a significant degradation of the cooperative search efficiency. In all trials in this experiment, the human search was started first. The robot waited for 40 seconds and then joined the human search. The search efficiency with explicit and implicit interactions are given in Fig. 9. It can be seen that both explicit and implicit cooperations outperforms the non-cooperative search. The average time cost with implicit interactions is only 17s (13.4%) higher than that with the explicit interactions. The dash line in Fig. 9 represents the actual driving time of the robot. It shows a byproduct with using implicit cooperation that it saves the driving and navigation cost of the robot, which is very energy consuming.

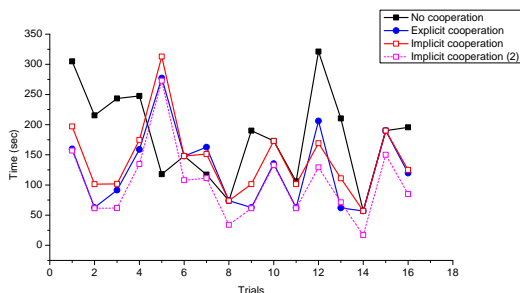


Fig. 9. Comparison in search efficiency

V. DISCUSSION AND CONCLUSION

In this paper we present a multi-agent cooperation scheme for search tasks. The robot agents infer the current state and intention of the human peer using human search activity model that acquired in the learning session. By inferring human state, the robot chooses complementary actions to achieve efficient cooperation with human peers. An implicit interface design for robot assisted tasks is also proposed in this paper, which allows the robot to infer the intention of the user and to provide assist autonomously. It reduces the cognitive workload of the user and therefore is useful for

elder care applications. The experimental results show that our approaches are effective and efficient.

In this paper, we focus on investigating the receptive behavior of the robot, where the human subject is not highly sensitive to the behavior of the robots. An example scenario is a human subject and embedded in an array of stationary surveillance cameras. However, our results show that this models also applies in applications where the behavior of the human subject can be altered by the robot's presence (e.g., H-R cooperation in a rescue-search team). The same model makes relevant predictions about actions that improve the performance of the team when informed by the two-agent search history. It does not depend on which agents contributed to the history given the assumption that the human and robot have similar observation capabilities. In the future, we plan to evaluate the added value (versus cost) of modeling joint team activity on the performance of search tasks.

In current stage we only evaluated the system with simulation experiments. Our group also has experience with using physical robot in human-robot cooperative search. In the near future, more experiments will be performed on the physical robot. The application of implicit interface on other daily activities such as reading and watching TV will be investigated with real world experiments. The performance of our approaches when the number of objects and activities scale up will be evaluated. Future work also includes using reinforcement learning to learn the expressive behaviors of the robot to naturally communicate with human.

REFERENCES

- [1] Intel open source computer vision library.
- [2] <http://talos.cs.umass.edu/yunlin/search/video/MURIDemo.m4v>.
- [3] T. Carlson and Y. Demiris. Human-wheelchair collaboration through prediction of intention and adaptive assistance. ICRA 2008.
- [4] P. Deegan, R. Grupen, A. Hanson, E. Horrell, S. Ou, E. Riseman, S. Sen, B. Thibodeau, A. Williams, and D. Xie. Mobile manipulators for assisted living in residential settings. *Autonomous Robots, Special Issue*, 2007.
- [5] T. Fong, C. Thorpe, and C. Baur. Multi-robot remote driving with collaborative control. *IEEE Tran. on Industrial Electronics* 50(4): 699-704, 2003.
- [6] S. G. Hill and B. Bodt. A field experiment of autonomous mobility: operator workload for one and two robots. HRI 2007.
- [7] C. A. Johnson, J. A. Adams, and K. Kawamura. Evaluation of an enhanced human-robot interface. *IEEE International Conference on Systems, Man and Cybernetics*, 2003.
- [8] M. W. Kadous, R. K.-M. Sheh, and C. Sammut. Effective user interface design for rescue robotics. HRI 2006.
- [9] R. R. Murphy. Human-robot interaction in rescue robotics. *Systems, Man and Cybernetics, Part C, IEEE Tran. on* 34(2): 138-153, 2004.
- [10] Y. Nevatia, T. Stoyanov, R. Rathnam, M. Pfingsthorn, S. Markov, R. Ambrus, and A. Birk. Augmented autonomy: Improving human-robot team performance in urban search and rescue. IROS 2008.
- [11] N. T. Nguyen, D. Q. Phung, S. Venkatesh, and H. Bui. Learning and detecting activities from movement trajectories using the hierarchical hidden markov model. CVPR 2005.
- [12] L. R. Rabiner. A tutorial on hidden markov models and selected application in speech recognition. *Proceedings of the IEEE* 77 (1989).
- [13] A. Tavakkoli, R. Kelley, C. King, M. Nicolescu, M. N. Nicolescu, and G. Bebis. A vision-based architecture for intent recognition. *ISVC* (2) 2007: 173-182.
- [14] D. Xie, R. Grupen, and A. Hanson. Context-aware search using cooperative agents in a smart environment. *IEEE Workshop on Applications of Computer Vision (WACV)*. December, 2009.