In: Proceedings of the 39th German Conference on Artificial Intelligence (KI2016), pp. 102-115, Springer International Publishing. Klagenfurt, AT, September 2016

A Robotic Home Assistant with Memory Aid Functionality

Iris Wieser(✉), Sibel Toprak, Andreas Grenzing, Tobias Hinz, Sayantan Auddy, Ethem Can Karaoğuz, Abhilash Chandran, Melanie Remmels, Ahmed El Shinawi, Josip Josifovski, Leena Chennuru Vankadara, Faiz Ul Wahab, Alireza M.Alizadeh B., Debasish Sahu, Stefan Heinrich, Nicolás Navarro-Guerrero, Erik Strahl, Johannes Twiefel, and Stefan Wermter

University of Hamburg, Department of Informatics, Knowledge Technology (WTM), Vogt-Kölln-Straße 30, D-22527 Hamburg, Germany
{4wieser,heinrich}@informatik.uni-hamburg.de

Abstract. We present the robotic system IRMA (*Interactive Robotic Memory Aid*) that assists humans in their search for misplaced belongings within a natural home-like environment. Our stand-alone system integrates state-of-the-art approaches in a novel manner to achieve a seamless and intuitive human-robot interaction. IRMA directs its gaze toward the speaker and understands the person's verbal instructions independent of specific grammatical constructions. It determines the positions of relevant objects and navigates collision-free within the environment. In addition, IRMA produces natural language descriptions for the objects' positions by using furniture as reference points. To evaluate IRMA's usefulness, a user study with 20 participants has been conducted. IRMA achieves an overall user satisfaction score of 4.05 and a perceived accuracy rating of 4.15 on a scale from 1-5 with 5 being the best.

Keywords: Robotic Home Assistant · Human-Robot Interaction · Social Robotics · Memory Service System · Speech Recognition · Natural Language Understanding · Object Detection · Person Detection

1 Introduction

Trying to find misplaced belongings may be time consuming and might end in frustration. A study about domestic assistive systems has shown that older adults would prefer robotic assistance over human help to support them in finding lost objects at home [3].

One assistive system developed for this task is the *Home-Explorer* presented by Guo and Imai [16]. It locates objects, that are equipped with smart sensors, in an indoor environment and is operated by a search interface. Deyle et al. [10] use a similar approach by attaching RFID (*Radio-Frequency Identification*) tags to household objects, which can then be found by a robot. Another example is the robotic home assistant *Care-O-bot 3* presented by Graf et al. [15], that can execute fetch and carry tasks on objects. The user selects the object using a touch screen attached to the robot.

For such robots to be incorporated into everyday life, however, additional aspects beyond functionality need to be considered. Foster et al. [13], for example, present a robot bartender that can operate in dynamic social environments. They identify both task success and dialogue efficiency as the main factors contributing to user satisfaction. Fasola and Matarić [11] present a robotic system that engages elderly people in physical exercise and conclude that users strongly prefer a physical robot embodiment instead of a computer simulation. To our knowledge, no working object finding system exists that provides a physical robot embodiment, offers a natural and intuitive interaction, and is independent of external sensors (e.g. on objects).

In a student project, we developed the stand-alone robotic system IRMA (*Interactive Robotic Memory Aid*) that can help users find various objects in an indoor home-like environment. IRMA integrates the required functionalities in a stable and robust manner, aims for a more intuitive and natural interaction, and is capable of learning the position of objects without the support of external hardware. This paper presents system details and the scores IRMA received in a user study. Also, the aspects of the system that have an impact on the users' opinions as well as further insights gained in the study are discussed here.

2 The IRMA System

IRMA is a domestic robotic system that assists people in their search for misplaced belongings.[1] It provides help in two ways, either by *moving* to the position of the requested object or by *describing* the requested object's position using other objects in the scene as reference points. The robotic system is able to navigate through the home environment in a collision-free manner. To do so, the robot creates a map beforehand. Knowledge about the current positions of all objects is acquired by performing an initial exploration run through the environment, during which the objects are detected and located on the map.

2.1 Architecture

We implemented IRMA as a distributed system in ROS (Indigo) [25]. As shown in Fig. 1, the overall system is decomposed into eight modules which can be grouped into four categories:

- ***Communication:*** There are three communication-related modules in the system. The *Speech Recognition* module recognizes human speech and converts the audio input into a string representation. The *Natural Language Understanding* module takes the string as input and identifies the desired object and type of action, which can be either "move" or "describe". The *Speech Production* module allows the output of generated natural language descriptions, e.g. to describe an object's position.

[1] A video showing the robot's performance is presented in the video session of the IEEE RO-MAN 2016 conference [29].

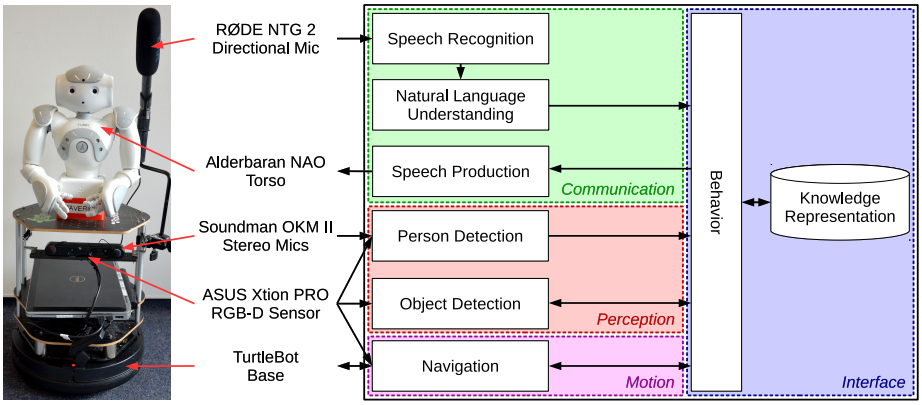

Fig. 1. An overview of the IRMA system: A picture of the robotic platform is shown in addition to a list of the used hardware components *(left)* and the decomposition of the IRMA system *(right)*. The arrows depict the data flow.

- ***Perception:*** Two modules provide the required perceptual capabilities. The *Object Detection* module uses visual input to detect and locate relevant objects in the scene. The *Person Detection* module uses both visual and audio input to do the same for persons.
- ***Motion:*** *Navigation* performs exploration in an environment, maps it and uses the map to navigate through the environment without any collisions.
- ***Interface:*** The *Behavior* module realizes the interface between all modules. It is the core of the system and contains the control of the robot's behavior. For storing knowledge it relies on the *Knowledge Representation* module that provides a database. It is also responsible for generating natural language descriptions of an object's position relative to other objects in the scene.

2.2 Robot Platform

IRMA's robotic platform is composed of different hardware components, shown in Fig. 1. The base component is a NAO torso. It offers a significant number of in-built functionalities such as turning text into speech used by the *Speech Production* module. Also, its appearance is very likely to make the overall system look more approachable, as shown in [26]. The NAO torso is mounted on a TurtleBot platform. The TurtleBot is accessed and controlled by the *Navigation* module. To get both, depth information and high quality RGB images, an Xtion camera is used, which is attached to the TurtleBot. The Xtion camera is used by the *Object Detection*, *Person Detection* and *Navigation* modules. The robotic platform is also equipped with a directional microphone and stereo microphones. As the robot faces the human during a conversation, a directional microphone enables a more robust speech recognition by reducing the noise from different directions. The stereo microphones are used to perform sound source localization to determine the position of the person relative to the robot [24].

2.3 Methods

Speech Recognition: To convert speech signals to textual transcriptions the speech recognition framework DOCKS (*Domain and Cloud-based Knowledge for Speech recognition*) [27] is adapted. The concept behind DOCKS is to combine the recognition advantages of large-scale ASR (*Automatic Speech Recognition*), here Google ASR, with phoneme-based post-processing techniques. This restricts the very general cloud-ASR results to a more specific, domain-based language.

To generate the domain-based language, user data has been collected via an online form. The users were asked to write down sentences to make the robot execute a task-object combination. These phrases are used as domain-specific hypotheses. The hypothesis with the lowest phonetic Levenshtein distance [19] to any of the cloud-ASR results is selected as the final textual transcription.

Natural Language Understanding: To understand the user command, the following semantic words need to be identified: The requested action, the object of interest and corresponding attributes (e.g. "find", "ball", "red"). Filtering keywords is straightforward and fast, but it is restricted to specific words and is highly error-prone (e.g. "I will find the ball" has a different meaning than "Can you find the ball?"). Other known approaches, such as semantic role labeling [23], rely on hardly accessible corpora that do not focus on the grammatical constructions required in our scenario (e.g. direct and indirect questions).

Thus, a combination of bi-gram scoring, an ESN (*Echo State Network*) based on Hinaut et al. [17], and a filter is utilized, see below. The ESN has been modified to extract also attributes and special "clues". The clues are important to differentiate the meaning of sentences like "Tell me the *color* of the ball" and "Tell me the *location* of the ball". Thus, the roles extracted by our modified ESN are `predicate(object,clue)` and `object(attribute)`, e.g. `tell(ball,location)`, `ball(red)`. We chose to use an ESN with 750 reservoir units and a leak rate of 0.2 after empirically evaluating different numbers of reservoir units and different leak rates in a 6-fold cross-validation. In a pre-processing step, collocated words are detected using bi-gram scoring and joined to provide only a one-word representation to the ESN (e.g. "milk carton" to "milkcarton"). The filter that is additionally applied to the output of the ESN serves two purposes: The assurance that only context-relevant verbs and objects are extracted (e.g. "Where is my love?" is beyond the scope) and the recognition of predefined synonyms.

As each sentence is processed individually by the ESN, the system is not able to recognize context spanning more than one sentence. Also, it cannot handle collocations consisting of more than two words or anaphora. Despite that, with the proposed approach, IRMA is capable of recognizing a substantial number of different grammatical constructions. Compared to grammar-based approaches, the user is not limited to a specific set of commands, but can use various sentences such as direct and indirect questions as well as imperative statements. This gives the user freedom in formulating a request intuitively and naturally. While the false negative rate lies above 93%, it achieves a true positive accuracy rate of 82% on an independently collected test set.

Object Detection: A pipelined approach with classical image processing techniques is used to detect and locate objects in real-time: First, region proposals are extracted. Mean shift filtering [8] is used to smoothen the image. The image is then binarized using adaptive thresholding and contours are extracted. The bounding boxes around the contours define our ROIs (*Regions Of Interest*). Due to the nonparametric nature of the segmentation pipeline, detections are independent of specific scenarios and objects.

In the next stage, SIFT (*Scale-Invariant Feature Transform*) [21] features are extracted from each ROI and BoW models (*Bag of Visual Words*) are created for each ROI [12]. These are vectors counting the occurrence of certain groups of features that are listed in a codebook. The codebook is constructed beforehand by extracting SIFT features on all training images and performing k-means clustering on the concatenation of the features. BoW models destroy the spatial structure of the features that constitute an object, therefore we employ a spatial pyramid-based ROI representation [18] to partially retain that structure: Each ROI is recursively decomposed into four cells, where the depth of the recursion is three. For the cells on the same layer, the BoW models are constructed and are concatenated to obtain a layer-based intermediate representation. The final ROI representation is then obtained by concatenating the vectors for all layers.

In the final stage, the ROI representations of all training images are used to train K-SVM (*Kernel Support Vector Machine*) classifiers [9]. To deal with the high-dimensionality in the histograms representing the ROIs, Histogram Intersection was used as a kernel distance metric [1].

To reduce the number of false positives caused by a noisy background, "object background classes" are created. These classes act as a buffer between the object and the background class in the dataset. This approach is combined with median filtering on the list of detections.

Person Detection: To detect and then locate human presence in the robot's surroundings, visual as well as audio input is used. A pre-trained OpenCV Haar features-based cascade classifier is applied to the image for frontal face detection [28]. This classifier can yield multiple face candidates, among which the one with the largest bounding box area is selected. As the visual field alone is limited, sound source localization is additionally performed on audio input coming from the stereo microphones. The implementation is based on Parisi et al. [24], where the angle of the sound source relative to the robot is estimated using TDOA (*Time Difference Of Arrival*) [22]. While it is not possible to distinguish between a sound source that is located in the front or in the back, it can be determined whether the sound source is located to the left or to the right, in a range of $\pm 90°$. This was taken into consideration when implementing person tracking: If no face is detected in the visual field, but a sound is located, IRMA is turned towards the sound source incrementally based on the sign of the angle value. It stops turning once a face has been detected or a maximum number of turns has been performed.

Navigation: Robot localization and exploration of the environment rely on the `navigation` stack of the ROS middleware. Robot localization is achieved using the `amcl` stack, which uses a particle filter to track the pose of a robot against a known map (Adaptive Monte Carlo Localization [14]). The robot explores its environment by navigating in a collision-free manner to a sequence of waypoints distributed throughout the room. If a particular waypoint is unreachable (e.g. because of an obstacle), the robot drops the unreachable waypoint and carries on to the next. The aim of exploration is to identify known objects and locate their positions on a two-dimensional map of the environment. To do so, the robot first identifies objects using the *Object Recognition* module. The centroid of the object in the depth image is utilized to calculate the three-dimensional coordinate of the object with respect to the robot's reference frame. This 3D coordinate is converted to a 2D coordinate and stored in the knowledge database for later usage. To allow the robot to continuously move and, at the same time, process a given frame for the object recognition task, timestamps are used to query caches of transformations and depth information.

In order to move to a particular object, the object's position needs to be retrieved from the knowledge base and a valid path needs to be calculated. However, as objects are usually placed on or even inside furniture (e.g. in a shelf), the object's position itself cannot always be chosen as the final destination point for the robot. To overcome this problem, goal rectification was implemented. This process results in a new goal, which is as close as possible to the original goal (if the original goal is within an obstacle), and ensures that the goal is reachable for the robot. The final orientation of the robot is chosen so that it faces the object when it reaches its goal.

Knowledge Representation: Knowledge about the environment (e.g. object positions) is stored in the RDF (*Resource Description Framework*) format. The description of an object's position is generated with respect to the robot's viewpoint. This verbal description is calculated in two steps.

At first, the reference objects that are closest to the requested object are determined based on the Euclidean distance. If the two closest reference objects have approximately the same distance to the requested object and it lies within their convex hull, the requested object is considered to be "in between" both.

If there is no such relation, the direction ("right", "left", "front", "behind") of the closest reference object needs to be computed. Initially, the perspective is normalized so that the requested object becomes the center of the new coordinate system and the y-axis corresponds to the robot's viewing angle. The Cartesian quadrant, in which the reference object (that is, its representative point) is, determines two possible directions. In Fig. 2 (left) for example, the representative point is in Quadrant I, so the directions are "front" and "left". The reference object's bounding box is used and a 45°-diagonal is computed through the corner that points towards the opposite quadrant. The side of the diagonal, on which the requested object is, determines the actual direction. In the example, it is below the diagonal, and thus the requested object is "in front" of the reference object. In Fig. 2 (right) a visualization of generated descriptions is shown.

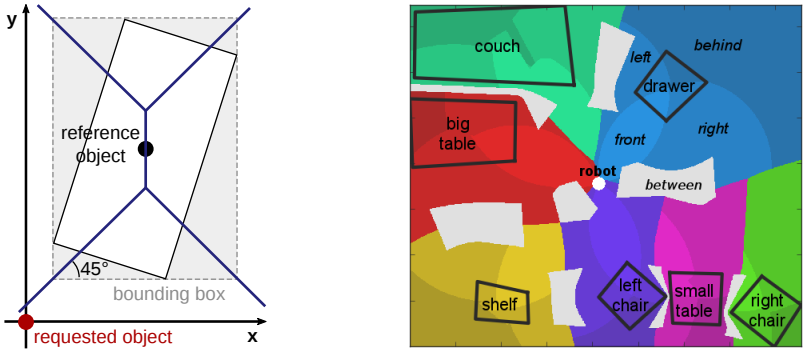

Fig. 2. *Left:* Determining the spatial relations between objects. Here, the requested object is "in front of" the reference object. *Right:* The descriptions produced for a test environment (the robot in the center). Each colored area has the same reference object and the intensity corresponds to one of the four spatial relations, which are only shown for the drawer. Gray stands for "in between".

Behavior: The desired system behavior was modeled with SMACH [4], which is a library for designing complex task-level executives. SMACH is faster than other imperative scripting approaches or model-based task planners [5] and can be used by a task planning system as a procedure definition architecture. Here, four parallel state machine containers were implemented. Each container performs one of the following tasks: *Exploration*, *Object and Person Detection*, *Person Tracking* and *Object Finding*. Among the termination policies that SMACH provides to overrule an active state machine container by another one, preemption is used.

3 Evaluation

To evaluate the usability of IRMA, we conducted a user study[2] with 20 participants (8 female, 11 male, 1 not specified) of different nationalities and ages ranging from 20 to 50 years. The participants were proficient in English and their previous experience with robots varied significantly: 5 did not have any experience with robots, 10 had little experience and 5 were familiar with robots.

3.1 Experimental Setting

The user study took place in a living room environment, shown in Fig. 3. It consists of a couch, a table, a drawer and a shelf placed along walls and a setting of two chairs with a coffee table between them. The setting covered $3.7\text{m} \times 4.8\text{m}$. To provide consistent lighting conditions, all windows were shaded and artificial light was used. The room was mapped a priori to provide information about the layout and positions of static objects (i.e. furniture). Two objects, a *milk carton* and a *trash can*, can be moved within the environment.

[2] Our dataset is available at `https://figshare.com/s/d949d3410df8db468f77` [30].

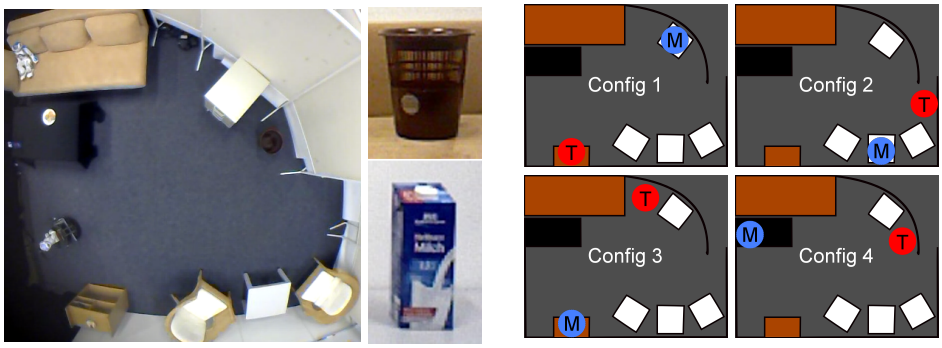

Fig. 3. *Left:* Setting and used objects. *Right:* Schematic map of all four object configurations used in the user study. The circle marked with the letter 'T' indicates the *trash can* and the circle marked with 'M' indicates the *milk carton*.

The participants were introduced to the environment and the robot IRMA. After being informed about the available objects and tasks that can be requested, they were asked to interact with the robot. In particular, the participants were asked to speak in a moderately loud voice and to repeat their command if IRMA does not react within 5-10 seconds.

In each user study session, a participant performed 8 runs in total, where each run ends with the robot performing the command. After the first 4 runs, the placement of the objects within the room was changed. All object configurations used in the study are shown in Fig. 3 (right). After each run, the participants were asked to rate how satisfied they were with the performed action and how accurate the system was in their opinion. After the complete session, the participant filled out a questionnaire comprised of three parts: (1) the SUS questionnaire [6], which measures overall usability, (2) the GODSPEED questionnaire [2], which measures five key HRI aspects, namely *Anthropomorphism*, *Animacy*, *Likeability*, *Perceived Intelligence*, and *Perceived Safety*, and (3) additional questions regarding the overall performance of the system that were answered on a 5-point Likert scale [20].

3.2 Results

Overall, IRMA correctly understood the requested type of help in 82.9% and the requested object in 92.1% of all runs performed during the user study. We did not consider two runs that had to be aborted and six runs that were not performed due to network issues. The response time, which was measured for each individual run, is the time between the end of the user's request until IRMA finishes its task, i.e. either having moved in front of the object or having finished its verbal description. For all runs without repetitions, the average response time for the task *describe* is 10.1 seconds, whereas the average response time for the task *move* is 36.8 seconds. For the *move*-task the robot did not move in 19.5% of the runs since it was already positioned close enough to the requested

object. In the other cases the distance to the object was reduced, except for one run (out of 82 in total). The results of the user feedback are summarized in Tab. 1. IRMA achieves a mean SUS score of 77.3, which translates to a C on the *Grade Scale* and to a *Good* on the *Adjective Rating Scale* according to [7]. The achieved GODSPEED scores, which evaluate key HRI aspects, are shown in Tab. 1. IRMA was perceived as likeable by the participants (4.28). However, it received a comparatively low score for antropomorphism (2.95).

Table 1. User study results including the SUS score (on a scale $0-100$ with 100 being the best), the GODSPEED scores, and the scores computed from the additional questions on a scale of $1-5$ with 5 being the best score

SUS	Mean (±StD)
Score	77.3 (±15.3)
GODSPEED	Mean (±StD)
Anthropomorphism	2.95 (±0.66)
Animacy	3.19 (±0.70)
Likeability	4.28 (±0.57)
Perceived Intelligence	3.61 (±0.69)
Perceived Safety	3.86 (±0.48)

Additional Questions	Mean (±StD)
Average over all runs	
User Satisfaction	4.05 (±0.56)
Perceived Accuracy	4.15 (±0.56)
Average over all sessions	
Usefulness for Elderly	4.15 (±1.19)
Intuitiveness	4.25 (±0.94)
Enjoyment	4.21 (±0.83)

The mean user satisfaction is little correlated with the perceived quickness of response (correlation value of 0.35). No correlation could be found between the final distance of the robot from the queried object and the satisfaction (0.0) or accuracy (-0.13) perceived by the user for a *move*-task.

The users' satisfaction and assessment of accuracy are consistent for all four object configurations used, as shown in Fig. 4 (left). The standard deviation for the settings are roughly the same and they overlap across all settings. This indicates further that no setting was significantly better or worse than the others. The slightly lower value in the accuracy rating for Config 4 is most likely due to the central position of the milk carton, see Fig. 3 (right). The central position of the *milk carton* might make the usefulness of the system seem less valuable due to the obvious placement of the object. Also, after completion of the *move*-task, the robot was on average further away from the milk carton, whereas it got very close to the trash can. This interpretation can be justified by Fig. 4 (right) which shows lower accuracy ratings for the runs including the *milk carton* compared to the runs including the *trash can*, especially for the *move*-task.

Although the participants were informed in advance about the possible tasks IRMA can perform, in 5.2% of all runs users used commands like "bring" or "get me" to instruct the robot. 3.6% of all sentences include other object references (e.g. "Is the milk carton left or right from you?" or "Is the milk box on the table?"). Also, in 1.3% two tasks were requested within one sentence and in 1.9% anaphora were used. In total, in 13.5% of all sentences, the user asked IRMA for something that it was not able to understand or perform.

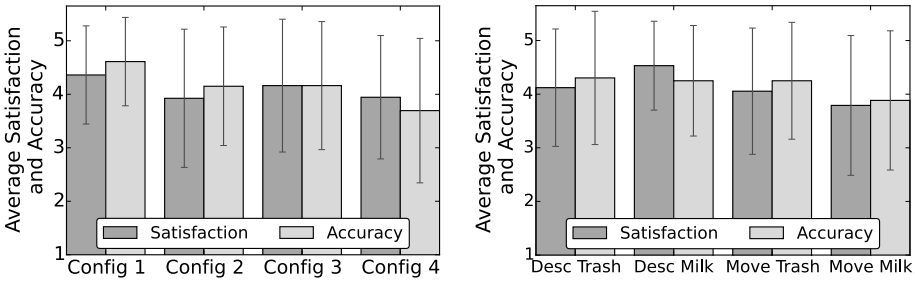

Fig. 4. *Left*: User satisfaction and accuracy scores for each object configuration *Right*: User satisfaction and accuracy scores for each individual task in Config 4

The number of times a user had to repeat a phrase until IRMA understood the command had a negative impact on his satisfaction, as shown in Fig. 5 (left). IRMA understood 50.7% of the instructions on the first try, while only 11.4% of the instructions had to be repeated more than two times. However, the assessment of accuracy does not seem to be affected by the number of repeats.

Fig. 5 (right) shows that the subjective rating of IRMA's intuitiveness as well as the user's enjoyment increased with how often the robot identified the task and object requested by the user correctly. The relation between the number of utterances that IRMA misinterpreted and the resulting intuitiveness (significantly worse only for 3 sentences) and enjoyment scores can be seen in this plot. The number of misinterpreted commands is correlated with intuitiveness and enjoyment, with the correlation coefficients being -0.54 and -0.51 respectively. It also turns out that the correct object being identified by the robot is more important for the satisfaction of the users than the intended task being performed by the robot. While the correlation coefficient for user satisfaction and correct object is 0.57, the correlation for user satisfaction and correct task

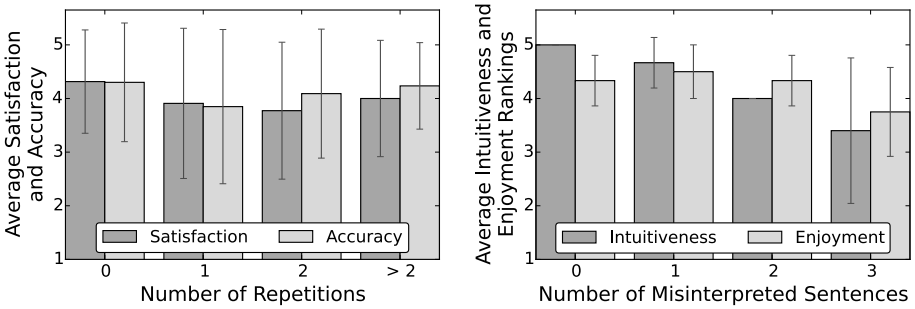

Fig. 5. *Left*: User satisfaction and accuracy scores versus the number of times a user had to repeat himself until a reaction of the robot was observed *Right*: The number of sentences misunderstood by IRMA versus the intuitiveness and enjoyment scores

is only 0.29. Similarly, the users perceive the performance of the robot as more accurate if the correct object is being identified compared to the correct task being performed. Here, the correlation coefficient for the perceived accuracy and the correct object is higher than the one for accuracy and correct task, with the values being 0.56 and 0.25 respectively.

Tab. 2 shows that all subjective factors obtained in the user study, such as satisfaction and accuracy, are uncorrelated to the previous experience the participants have had with robotics. The highest correlation value with experience with robots exists for a GODSPEED aspect, namely animacy, with still a low value of −0.29.

Table 2. Results of the correlation between the users' prior experience with robots and SUS, GS (GODSPEED) and additional subjective factors

Correlation	Experience	User Satisfaction	−0.28
SUS:Score	0.22	Perceived Accuracy	−0.25
GS:Anthropomorphism	0.02	Usefulness for Elderly	0.06
GS:Animacy	−0.29	Intuitiveness	0.08
GS:Likeability	0.10	Enjoyment	0.09
GS:Perceived Intelligence	−0.20	Perceived Quickness	−0.09
GS:Perceived Safety	0.02	Time Acceptable	0.00

4 Discussion

In general, IRMA performs well and achieves a high user satisfaction. However, there are certain parts of the system that can still be improved in future studies. The rotation of the robot was confusing to many participants. Firstly, *Person Detection* sometimes recognized false positives in a very cluttered environment and thus, the robot stopped rotating at the wrong time. Secondly, participants were not aware that the robot is trying to locate the user and sometimes misinterpreted this behavior as a "reaction" to their request.

Also, sometimes participants used sentences containing anaphora (e.g. "I still cannot find the milk. Can you show *it* to me?"). This occurred in 1.9% of all utterances. The *Natural Language Understanding* module, however, is not yet capable of understanding anaphoric references or context spanning multiple sentences. On average only every second request of the participants was performed by the IRMA system. In many cases, *Speech Recognition* was not able to recognize the sentence correctly. This might be due to the microphone not being clearly directed towards the user, as person tracking stopped too early. Moreover, in 21% of all recognized sentences, the sentences contained either an object or a task that was different from what was actually requested by the user. The results show that a wrongly understood object has more impact on the user satisfaction than a wrongly chosen task (*describe* or *move*). This is because the

position of the wrong object is not of interest to the user, while performing the wrong task with the correct object is still helpful.

Most people preferred one of the capabilities over the other (40% *move*, 55% *describe*, 5% neither-nor), which shows that it is useful to have both capabilities in our scenario. However, the fact that users often expected the robot to interact more with the environment and bring the object directly to them, indicates that this additional task might be a more helpful form of assistance to the user when compared to the *move*-task alone. The assistance could further be enhanced by other tasks, e.g. with a reminder functionality for taking medicine. One reason for the higher preference of the *describe*-task might be that the average response time for the *describe*-task took only about 25% of the time required for the completion of the *move*-task. Additionally, description tasks were rated with a higher accuracy (4.3) compared to movement tasks (4.02).

The results of our user study show that the previous experience of the users with robots has little influence on how they rated the system. In combination with a relatively high satisfaction score of 4.05 (on a scale from 1-5 with 5 being the best), this indicates that our system is intuitive to use. Moreover, the average response time for a *describe*-task only takes 10.1 seconds. Assuming that a search by the user without external help would take longer than 10 seconds, IRMA can save valuable time and effort of the user locating misplaced belongings.

5 Conclusion

IRMA is a stand-alone robotic system designed to help people in finding lost objects. Several state-of-the-art methods and frameworks have been integrated to enable an easy, robust and natural human-robot interaction. IRMA has the ability to explore the environment, detect objects and remember their positions. It can also describe the location of objects using natural language and is able to move to a specified object, when the user asks to do so using natural phrases (e.g. direct and indirect questions as well as imperative statements).

The results of our user study with 20 participants show that the system is able to accomplish its task to an average satisfaction rate of the user of 4.05 on a scale from 1-5 with 5 being the best. IRMA is able to identify the intention of the user for every second sentence that has been naturally uttered by the participants, and perform the corresponding task successfully. As the average response time for a successful description of the object's position is 10.1 seconds on average, IRMA can save time for users, especially elderly, finding a misplaced belonging. IRMA has also shown to be intuitive to use, as the user's previous experience with robots has no influence on the subjective evaluation of the system.

Acknowledgments. The authors gratefully acknowledge partial support from the German Research Foundation DFG under project CML (TRR 169), the European Union under project SECURE (No 642667), and the Hamburg Landesforschungsförderungsprojekt.

References

1. Barla, A., Odone, F., Verri, A.: Histogram Intersection Kernel for Image Classification. In: International Conference on Image Processing (ICIP). vol. 3, pp. 513–516. IEEE (2003)
2. Bartneck, C., Kulić, D., Croft, E., Zoghbi, S.: Measurement Instruments for the Anthropomorphism, Animacy, Likeability, Perceived Intelligence, and Perceived Safety of Robots. International Journal of Social Robotics 1(1), 71–81 (2008)
3. Beer, J.M., Smarr, C.A., Chen, T.L., Prakash, A., Mitzner, T.L., Kemp, C.C., Rogers, W.A.: The Domesticated Robot: Design Guidelines for Assisting Older Adults to Age in Place. In: Annual ACM/IEEE International Conference on Human-Robot Interaction. pp. 335–342. HRI '12, ACM/IEEE (2012)
4. Bohren, J., Cousins, S.: The SMACH High-Level Executive. IEEE Robotics Automation Magazine 17(4), 18–20 (2010)
5. Bohren, J., Rusu, R.B., Jones, E.G., Marder-Eppstein, E., Pantofaru, C., Wise, M., Mösenlechner, L., Meeussen, W., Holzer, S.: Towards Autonomous Robotic Butlers: Lessons Learned with the PR2. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 5568–5575. IEEE (2011)
6. Brooke, J.: SUS - A Quick and Dirty Usability Scale. In: Usability Evaluation In Industry, pp. 189–194. Taylor & Francis (1996)
7. Brooke, J.: SUS: A Retrospective. Journal of usability studies 8(2), 29–40 (2013)
8. Cheng, Y.: Mean Shift, Mode Seeking, and Clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence 17(8), 790–799 (1995)
9. Cortes, C., Vapnik, V.: Support-Vector Networks. Machine Learning 20(3), 273–297 (1995)
10. Deyle, T., Reynolds, M.S., Kemp, C.C.: Finding and Navigating to Household Objects with UHF RFID Tags by Optimizing RF Signal Strength. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 2579–2586 (Sept 2014)
11. Fasola, J., Mataric, M.: A Socially Assistive Robot Exercise Coach for the Elderly. Journal of Human-Robot Interaction 2(2), 3–32 (2013)
12. Fei-Fei, L., Perona, P.: A Bayesian Hierarchical Model for Learning Natural Scene Categories. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). vol. 2, pp. 524–531. IEEE (2005)
13. Foster, M.E., Gaschler, A., Giuliani, M., Isard, A., Pateraki, M., Petrick, R.P.: Two People Walk into a Bar: Dynamic Multi-Party Social Interaction with a Robot Agent. In: ACM International Conference on Multimodal Interaction. pp. 3–10. ICMI, ACM (2012)
14. Fox, D.: Adapting the Sample Size in Particle Filters Through KLD-Sampling. The International Journal of Robotics Research 22(12), 985–1003 (2003)
15. Graf, B., Reiser, U., Hägele, M., Mauz, K., Klein, P.: Robotic Home Assistant Care-O-bot 3 – Product Vision and Innovation Platform. In: IEEE Workshop on Advanced Robotics and its Social Impacts. pp. 139–144. IEEE (2009)
16. Guo, B., Imai, M.: Home-Explorer: Search, Localize and Manage the Physical Artifacts Indoors. In: International Conference on Advanced Information Networking and Applications (AINA). pp. 378–385. IEEE (2007)
17. Hinaut, X., Petit, M., Pointeau, G., Dominey, P.F.: Exploring the Acquisition and Production of Grammatical Constructions Through Human-Robot Interaction with Echo State Networks. Frontiers in Neurorobotics 8, 16 (2014)

18. Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). vol. 2, pp. 2169–2178. IEEE (2006)
19. Levenshtein, V.: Binary Codes Capable of Correcting Deletions, Insertions and Reversals. Soviet Physics Doklady 10, 707 (1966)
20. Likert, R.: A Technique for the Measurement of Attitudes. Archives of Psychology (1932)
21. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
22. Murray, J., Wermter, S., Erwin, H.: Auditory Robotic Tracking of Sound Sources Using Hybrid Cross-Correlation and Recurrent Networks. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3554–3559. IEEE (2005)
23. Palmer, M., Gildea, D., Xue, N.: Semantic Role Labeling. Synthesis Lectures on Human Language Technologies 3(1), 1–103 (2010)
24. Parisi, G.I., Bauer, J., Strahl, E., Wermter, S.: A Multi-Modal Approach for Assistive Humanoid Robots. In: Workshop on Multimodal and Semantics for Robotics Systems (MuSRobS). vol. 1540, pp. 10–15. CEUR Workshop Proceedings (2015)
25. Quigley, M., Gerkey, B., Conley, K., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., Ng, A.: ROS: An Open-Source Robot Operating System. In: ICRA Workshop on Open Source Software. vol. 3, p. 6 (2009)
26. Rosenthal-von der Pütten, A.M., Krämer, N.C.: How Design Characteristics of Robots Determine Evaluation and Uncanny Valley Related Responses. Computers in Human Behavior 36, 422–439 (2014)
27. Twiefel, J., Baumann, T., Heinrich, S., Wermter, S.: Improving Domain-Independent Cloud-Based Speech Recognition with Domain-Dependent Phonetic Post-Processing. In: AAAI Conference on Artificial Intelligence. vol. 28, pp. 1529–1535. AAAI Press (2014)
28. Viola, P., Jones, M.: Rapid Object Detection Using a Boosted Cascade of Simple Features. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). vol. 1, pp. 511–518. IEEE (2001)
29. Wieser, I., Toprak, S., Grenzing, A., Hinz, T., Auddy, S., Karaoğuz, E.C., Chandran, A., Remmels, M., El Shinawi, A., Josifovski, J., Chennuru Vankadara, L., Wahab, F.U., Mollaalizadeh Bahnemiri, A., Sahu, D., Heinrich, S., Navarro-Guerrero, N., Strahl, E., Twiefel, J., Wermter, S.: A Robotic Home Assistant with Memory Aid Functionality (May 2016), `https://www.informatik.uni-hamburg.de/wtm/videos/VideoSubmission_UniHamburgWTM_RO-MAN2016.mp4`, video. IEEE RO-MAN 2016. Accepted.
30. Wieser, I., Toprak, S., Grenzing, A., Hinz, T., Auddy, S., Karaoğuz, E.C., Chandran, A., Remmels, M., Shinawi, A.E., Josifovski, J., Vankadara, L.C., Wahab, F.U., B., A.M., Sahu, D., Heinrich, S., Navarro-Guerrero, N., Strahl, E., Twiefel, J., Wermter, S.: Dataset for "A Robotic Home Assistant with Memory Aid Functionality" (May 2016), `https://figshare.com/s/d949d3410df8db468f77`