Oops, Something Is Wrong
Error Detection and Recovery for Advanced Human-Robot-Interaction

Thorsten P. Spexard, Marc Hanheide, Shuyin Li, and Britta Wrede

***Abstract*— A matter of course for the researchers and developers of state-of-the-art technology for human-computer- or human-robot-interaction is to create not only systems that can precisely fulfill a certain task. They must provide a strong robustness against internal and external errors or user-dependent application errors. Especially when creating service robots for a variety of applications or robots for accompanying humans in everyday situations sufficient error robustness is crucial for acceptance by users. But experience unveils that operating such systems under real world conditions with unexperienced users is an extremely challenging task which still is not solved satisfying. In this paper we will present an approach for handling both internal errors and application errors within an integrated system capable of performing extended HRI on different robotic platforms and in unspecified surroundings like a real world apartment. Based on the gathered experience from user studies and evaluating integrated systems in the real world, we implemented several ways to generalize and handle unexpected situations. Adding such a kind of error awareness to HRI systems in cooperation with the interaction partner avoids to get stuck in an unexpected situation or state and handle mode confusion. Instead of shouldering the enormous effort to account for all possible problems, this paper proposes a more general solution and underpins this with findings from naive user studies. This enhancement is crucial for the development of a new generation of robots as despite diligent preparations might be made, no one can predict how an interaction with a robotic system will develop and which kind of environment it has to cope with.**

I. INTRODUCTION

Within the last decades substantial progress has been made in robotic research now enabling systems not only to perform industrial and manufacturing tasks but taking more and more part in our daily lives. This produces new challenges for robotic systems, as the more they become part of our daily life, the less the situations and scenarios are predictable in which they have to operate. Unpredictable situations, however, are difficult for robots to manage as they impose unpredictable problems and errors.

In this paper we address the challenge of real-world applications imposed on robotic systems and present an approach to detect different interaction error cases and to recover and resume operation to continue HRI in a socially acceptable way.

In human-human interaction the communication partners manage to maintain a common-ground by explicit communication mechanisms as described in the grounding model by Clark [1] as well as by more implicit alignment strategies as proposed by Pickering and Garrod [2]. Yet, also in human-human interaction misunderstandings arise and need to be solved by the communication partners. While in such situations it is often sufficient to clarify one propositional fact as presented in[3], it sometimes happens that a complete interaction sequence has been interpreted differently by the interlocutors. In such an extreme case one strategy is to restart again from the beginning by explicitly “removing” the previous statements from the common ground. We present an approach based on this idea of resetting the interaction which can be triggered by both, the human user or the robot itself.

Fig. 1. BARTHOC Junior demonstrating a gesture for being puzzled by scratching the back of its head

While our system has been designed and evaluated on a mobile, non-humanoid robot our goal is to implement this on the humanoid robot *BARTHOC* (**B**ielefeld **A**nthropomorphic **R**obot for **H**uman **O**riented **C**ommunication) (Fig. 1) [4] in order to analyze the effects of anthropomorphism on the grounding and alignment strategies of the human user.

We will first start by describing the real word conditions for our robot in Section II. In Section III we demonstrate how the proposed error model is applied to the concrete scenarios comparing alternative approaches in error detection and recovery, followed by the concrete recovery implementation for errors in Section IV we observed during multiple interactions. We conducted two experiments in order to evaluate the error awareness and recovery system, first testing the error awareness and recovery explicitly with varying experienced users, and secondly analyzing the interaction capabilities of the robot during an evaluation with naive users in a real flat. The results are presented in Section V. We summerize our

This work has been supported by the European Union (COGNIRON project FP6-IST-002020) and the German Research Foundation (CRC 673)

The authors are with Applied Computer Science, Bielefeld University, D-33501 Bielefeld, Germany {tspexard, mhanheid, shuyinli, bwrede}@techfak.uni-bielefeld.de

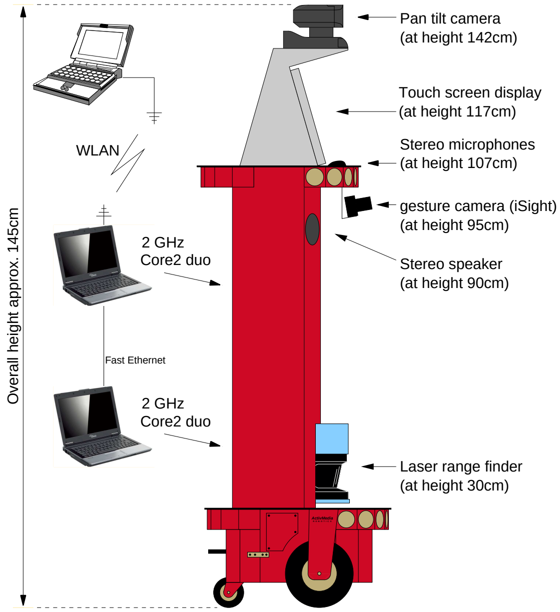

Fig. 2. Hardware composition of BIRON.

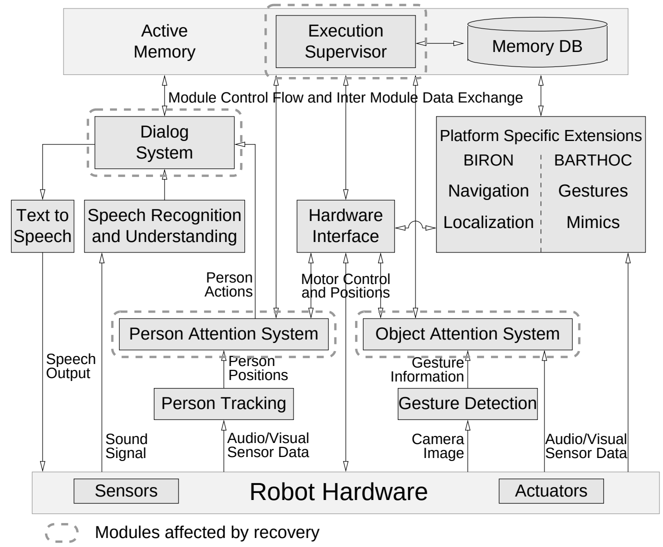

Fig. 3. Software architecture for the integration on the demonstrators BARTHOC and BIRON.

approach in a short conclusion in Section VI stressing the main achievements described in this publication.

II. APPLICATIONS IN THE REAL WORLD

In order to build robots that humans accept in their daily lives one needs to consider natural environments instead of lab situations. Since our goal is to port the proposed error recovery system on an anthropomorphic robot it is being developed on the basis of a generic architecture widely independent from specific demonstrators and environments. This software framework is also running on the anthropomorphic robot BARTHOC where we intend to carry out further experiments and developments.

A. Robot Platform and Environment

Our current platform is the mobile robot *BIRON* (**Bi**elefeld **Ro**bot Companio**n**) as shown in Fig. 2. It is equipped with several sensors that allow an assessment of the current situation as a basis for interaction and error recovery. (see Fig. 7). The scenario for which BIRON has been developed envisions a newly purchased robot being introduced to its new working area – usually an apartment – by the human user. Due to the huge variety of application areas especially in home environments only a small set of preprogrammed knowledge is useful. The major part such as maps or objects in the new environment has to be learned online during interaction with a person [5]. Thus, the Home-Tour-Scenario incorporates especially the requirement of a real-world environment with the additional constraint that the user has only minimal knowledge about the robot. For the testing and further development a real world apartment has been rented as a realistic testbed.

B. Implications

The complex hard- and software of our robot as well as the scenario have manifold implications for the interaction with the robots. Basically, we can differentiate between robot internal and external problem sources.

System Robustness: The software modules (see Fig. 3) have been developed by different partners at different institutes and were integrated into the system successively, each unveiling a different level of robustness, reliability, and performance. In such an integrated system, the precision and quality of the individual components will obviously affect the overall quality of the system. If the quality of the overall system was determined by the independent quality of the individual components, this would have dramatical consequences. If any error in the computation of one of the components would cause an error in the whole system that cannot be compensated, the quality would indeed be very poor, because the success rates of components are considered to be independent of each other in this case. In consequence, if every component exhibits a success rate of $P_i = 98\%$, the quality of an integrated system of $n = 30$ independently working components would statistically drop to $(P_i)^n \approx 54.5\%$.

Environmental Challenges: Experiments with naive users showed that unexpected situations occur surprisingly often leading to problematic robot behavior. One reason is that interaction with naive users can be problematic because the users do not know the capabilities of the system. But also physical conditions unforeseeable by the developers such as a sticky ground, blinding light sources or obstacles that can not be avoided while getting to a certain position provide sufficient situations for a robot to react in a way the user does not expect.

To enable a smooth continuation of an interaction we did not try to solve the Sisyphean task to identify and model all exceptional situations, but instead implemented different levels of error awareness and recovery.

III. MODELING ERROR-AWARENESS AND RECOVERY

Error awareness and recovery on the software level is crucial for autonomous robots acting in real environments with naive users as motivated before. However, error recovery has been studied mainly in the context of task execution failures.

E.g., within a situated module-based architecture an obstacle avoidance [6], prevents the robot from colliding during its navigation task. Laengel et al [7] classified operation errors of an assembly robot KAMRO and proposed recovery strategies for errors such as gripping wrong objects. Such predicatable errors and recovery strategies are explicitly represented in the dialog component of the system, which generates speech to report errors. Ross et al [8] classify system errors into four categories: anticipated errors, exceptional errors, irrecoverable errors and socially recoverable errors. They developed extra plans and commitment rules in their AgentFactory framework that enable an autonomous office assistant robot to request help from human users in difficult but predictable situations. This strategy provides the robot with relevant social skills to tackle problems such as existence of obstacles and is computationally less expensive than re-planning. The robot Grace [9], which can follow a person, is able to generate a pre-defined error message via speech when it loses track of the person.

As can be seen, existing work of error recovery in HRI mainly addresses operational errors of the system and requires the help of the user to solve the problem. Autonomous recovery by e.g. self-reconfiguration is not foreseen. We propose an error awareness and recovery model that systematically classifies interactional and operational errors in HRI (Fig. 4). This model is based on two dimensions concerning the *detector and initiative taker* and the *solution provider* of an error, respectively (y- and x-axis in Fig. 4). For instance, BIRON is able to detect the error if it loses track of the user during the interaction. In this situation, it takes take the initiative to ask the user to come back. Although the robot is able to *detect* the error, the final *solution* can only be provided by the user. In contrast, if the robot behaves in an unexpected way, this can by definition only be detected by the user, as s/he defines what 'unexpected' means. However, a solution can only be provided by the robot. "Mode confusion" describes a complex error pattern where the robot is in a different state as actually communicated to the user. This problem can be detected and solved by either the user or the robot. The more errors a robot can detect and solve itself, the more autonomous and usable it is considered to be.

On the basis of this model, we are able to identify different types of errors that should be handled by the robot interactively according to the following behaviors:

In the case of hardware defects, which occur sufficiently frequent in order to be modeled in the interaction framework, the robot should ask the user to contact a technician.

Speech recognition errors, disappearance of the user and existence of obstacles may at least partly be detected by the robot itself so that it can ask the user for cooperation. In case

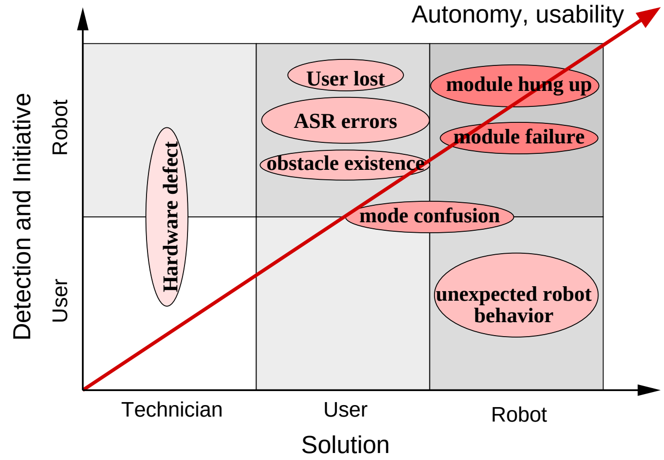

Fig. 4. Error awareness and recovery model (ASR = Automatic Speech Recognition).

of severe mode confusion or when the robot behaves in an unexpected way, the user has the possibility to ask the robot to reset the system explicitely via spoken commands to avoid further interaction problems. Similarly, the robot should be aware of failures of software modules or hanging up and solve these problems itself.

IV. INTERACTIVE ERROR RECOVERY

Due to the fact that our goal is *interaction* recovery, the major control has been implemented as part of the interface between the robot and the human user: the *dialog system*. In general, the dialog is responsible for carrying out interactions with the user including transferring user commands to the robot control system and reporting task execution results to the user. The dialog module is always kept up-to-date with the state of the robot via interaction with an internal system state machine. This knowledge enables the dialog to immediately be aware of possible problems of the entire robot system and to act accordingly. We made use of this advantage and implemented various advanced error recovery strategies in the dialog system [10], [11].

A. Error Detection and Recovery

Mode Confusion: To handle, e.g., mode confusions, the dialog system first estimates the performance of the user. For this purpose, the system counts illegally proposed tasks by the user. An illegal task is one that can not be performed by the robot in a certain robot state. Immediately after the user proposes such a task, the dialog system initiates presentations to inform the user of what s/he should do to achieve that task given the current robot state. This strategy is usually sufficient to help the user solve the problematic situation. However, as we observe in user studies, some users are still confused or the system still is in an unexpected state: the user continues to propose currently illegal tasks. When this happens more frequently than an adaptive threshold, – which is individually set based on the user's experience level and interaction preference – the dialog system initiates a presentation to ask the user whether the system should perform a reset. Furthermore, the user can explicitly trigger a recovery when s/he feels the current interaction situation to be out of control by saying “Reset!”. In these cases, the dialog triggers the execution supervisor to actually reset the respective modules in the system: Person Attention System, Object Attention System, Execution Supervisor, and the dialog system as highlighted in Fig. 3.

Module Hangups: Another severe problem are module hangups which require another autonomous error detection and recoverability ability of the robot. The realization of error recovery in this situation is based on the communication between the dialog system and the central control module Execution Supervisor. The dialog system receives “heart-beat” messages from the Execution Supervisor periodically. When this heartbeat slows down below a certain threshold the dialog system generates an utterance where it asks to user to wait. When the communication between Execution Supervisor and the dialog system can not be resumed after a certain amount of time, the dialog system generates another presentation informing the user about the break-down and asking the user to contact a technician (which in a real system this should actually be broken down to a user-initiated system reboot) (see Fig. 4).

Problems Perceived by Robot Sensors: In other error situations, the dialog system receives error messages from other modules, e.g., message from the Hardware Control of the robot about the existence of an obstacle, and generates appropriate presentations. This way, the robot informs the user of the reason why it can not move and asks the user for help (“Sorry, I’m stuck. Push me away manually, please!”).

V. EVALUATION AND DISCUSSION

Research with respect to interactive error detection and recovery demands for user studies in scenario-oriented tasks. In order to gather insights about the effectiveness and usefulness of our approach and to identify its limitations we conducted an initial pre-study with BIRON with a strong emphasis on the newly added error model. Subsequently a more complex study was conducted under real world conditions with naive users evaluating the system in a less constrained interaction.

A. Pre-Study

For the first study subjects were instructed to perform a typical home tour related task in a laboratory room. This task consisted of (i) an initial interaction with the robot, (ii) guiding the robot around, and (iii) telling it the name of the current room so that the robot could build a map of the apartment. The following script of an ideal run was used to instruct subjects:

U: Hello Biron.
R: Hello.
U: Follow me.
R: OK, I follow you.
U: Stop.
R: OK, I stopped.
U: This is the living room.
R: OK, nice living room.
U: Biron, follow me.
R: OK, I follow you.
U: Biron, stop.
R: OK, I stopped.
U: Biron, where are you?
R: This is your nice living room.

It should be noted that an ideal run is unlikely to happen in real worlds scenarios, as users typically vary their interaction patterns or the system makes errors in its perception and interpretation. Hence, all deviations from this ideal case have been recorded by observing subjects during their performance to complete the given task.

The results of this study are summarized in Fig. 5. The number of occurrence of selected interesting error patterns and the different modes of recovery are shown in the figure for each experimental run. Furthermore, we measured the overall time of each run from first contact with the robot to leave-taking after completion. The runs have been conducted with three users of different experience levels. Each user conducted the task twice. Although the number of runs (six) is rather small in this preliminary study, the effects and benefits already become apparent from these results. The least-experienced user (runs 5 and 6) used the system for the first time, while the most experienced one (runs 1 and 2) was one of the developers. This becomes also apparent when looking at the overall time per run which is the shortest for the experienced user (run IDs 1 and 2) who knew what to do most of the time from habituation effects.

As a general positive result, it has to be noted that all participants of the study have been successful in completing the given task in reasonable time. However, none of the runs has been carried out perfectly and errors dedicated to the interactive situation occurred as shown in Fig. 5. The column “Repeated Commands” summarizes the number of events, where the robot did not react immediately to the user’s request, requiring a repetition of the utterance. The column “user lost” shows the number of cases, where the communication with the robot had to be re-initiated because the system’s perception had lost track of the user. Most of these events result from tracking losses of the Person Attention System system during the robot’s turn to acquire a spatial model of a room (“This is the living room.”). All users succeeded to immediately recover interaction with the robot in these cases by following the help message provided by the system once it had detected the loss of the interaction person.

Even more interesting are the numbers of unexpected behaviors of the robot during the task. A typical example for an unexpected robot behavior is the case where the user asks the robot to follow her, the robot acknowledges this request, but does not actually follow. The reasons for unexpected system behavior can be various and should be analyzed in detail. However, the explicit interaction recovery allowed the subjects to solve the communication problem without additional explanation or external help from the experimenters, as reflected in the column “robot initiative”. If the system did not recognize the inconsistent system state, explicit recovery

ID	Repeated commands	Error pattern: does not behave as expected	user lost	understanding errors	recovery / reset: robot initiative	user initiative	situation-related help	Time/run
1	5		1	1				00:02:50
2	1	1	1	7	1	1		00:03:20
3	2	1	2	4	1	2	1	00:06:30
4		1	1	3		2		00:03:00
5	2	1	1	10		3	1	00:06:20
6	2		1	3			2	00:03:20

Experience

Fig. 5. Quantitative results from the prestudies.

by the user was necessary as indicated by column "user initiative". As can be seen the least experienced user used this features most often (run 5 and 6). Another promising finding, which is also substantiated by our every day experience with the robot, is that repeated interaction problems between the system and the user due to mode confusion (summarized under the term "understanding errors") lead to a high number of illegal task requests in the dialog and in consequence to recovery initiated by the robot itself.

However, mode confusion can also be solved by situation-related help. The eighth column ("situation related help") summarizes the number of such help initiatives of the robot triggered due to the currently illegal task requested by the user. Especially from the last run (ID 6) one can conclude that this help successfully solved the misunderstanding cases as no reset was necessary at all.

B. *Real-World Conditions*

Taking the encouraging results of the prestudy into account, a new evaluation for the complete system was set up. Taking advantage of free access to a flat ten naive users who did never interact with the robot before were invited. Following the home tour scenario the users had the task to guide the robot from its starting position in the living room trough the hallway to the dining area as depicted in figure 6. Subjects were also asked to teach the robot the names of the different rooms and to show easy to move objects like cups and furniture like tables or armchairs to the robot while moving around (see Fig. 7).

As our overall goal is an intuitive usage of the robot we want to prevent the user from reading complex and less inspiring manuals. Therefore in our scenario only a brief introduction to the robot was given to the user by an experienced user, lasting from 5 up to 10 minutes. During this introduction the subjects were able to ask questions or try out different aspects of the system. Subsequently to the introduction the experiment started by giving a list of tasks to the subject which should be performed while proceeding to the dining room:

1) Show the living room to the robot.
2) Show the armchair to the robot.
3) Proceed with the robot to the dining room.
4) Show the dining room to the robot.
5) Show the table to the robot.

Compared to the prestudy this task list did not include concrete commands and so enabled the subjects to interact more freely with the robot. To achieve comparable experimental conditions and avoid weariness of the participants a time limit for the second part of the experiment of 15 minutes per subject was set.

We want to stress that all subjects, except one, successfully completed the task within time although they had to interact with a complex robotic system in an unstructured environment without a fixed script. The user who was not able to complete the task had already proceeded with BIRON to the hallway but ran out of time shortly before s/he could complete the experiment. Focusing especially on the role of system recovery, in seven out of the ten runs the error model was explicitly used by the subjects and in every run the error awareness of the robot helped the human to solve, e.g., collision avoidance in cooperation. Table I presents detailed information for each subject, demonstrating that 25% of the user-initiated error recovery corresponded to repeated classification errors of the speech recognition. But 75% of the user-initiated resets were based on a robot behavior that was not expected nor understood by the user in a particular situation. This supports our approach to immediately regain common ground. Even though system designers should focus on minimizing such situations they will not be able

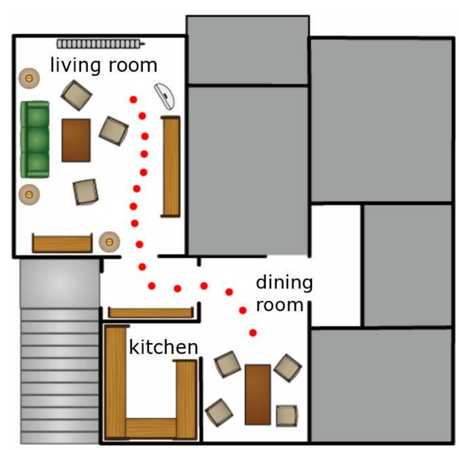

Fig. 6. Path taken from user and BIRON in the flat. The experiment started in the living room and BIRON followed the user into the dining area, while the user showed locations and objects to the robot.

(a) Guiding the robot.

(b) Introducing a *room*.

(c) Introducing an *object*.

Fig. 7. The home tour with BIRON.

to exclude them at all. In our case the average use of an explicit reset was 1.2 times per subject and run which demonstrates that the presented system already achieved a robust interaction quality. Note that without the proposed error model the overall success rate of the evaluation of 90% would not have been realizable. Even more, some of these errors might have lead the user to give-up the interaction as it is not unlikely that in the further interaction no way to re-establish common ground could be found. These results show that error awareness and recoverability at runtime are important features in order to enable a smooth human-robot interaction.

VI. Conclusion

We have shown an approach to deal with unpredictable interaction situations by decomposing the problem into two dimensions: the detection and communication of a failure and the provision of a solution. While many known error recovery strategies only take solutions provided by the user into account, we present a model that explicitly involves a deep analysis of the situation by the robot so that it may provide a solution by itself.

A first evaluation of this model indicates that (1) in every interaction error-situations occur (2) unexpected robot behavior tends to correlate with user triggered resets, indicating the usefulness of this feature (3) understanding errors lead to robot-triggered recovery or help suggestions and (4) by these help strategies all subjects were able to cope with the problem without external help. In a second more sophisticated evaluation the error model proofed its usability in assisting naive users in a complex interaction scenario such that nine out of ten subjects successfully accomplished the given tasks with the tenth person nearly succeeding but exceeding a given time limit.

Our future work is guided towards porting this system on a humanoid robot in order to analyze the effects of anthropomorphism on the communication of internal and external error sources as a basis for error recovery in human-robot interaction.

ID	user reset	repeated commands	unexpected behavior	robot assist.	time / run
1	2	1	1	1	09:03
2	2		2	2	aboard
3	3	1	2	2	10:40
4				2	05:34
5	1		1	2	12:21
6				1	06:47
7	1		1	1	09:16
8	1	1		1	06:54
9	2		2	1	08:33
10				2	7:41
⌀	1.2	0.3	0.9	1.5	8:32

TABLE I

System recovery: User initiated resets and robot initiated requests for assistance supported the subjects during evaluation.

References

[1] H. H. Clark, Ed., *Arenas of Language Use*. University of Chicago Press, 1992.

[2] M. J. Pickering and S. Garrod, “Toward a mechanistic psychology of dialogue,” *Behav Brain Sci.*, vol. 2, no. 27, pp. 169–190, Apr 2004.

[3] M. Purver, “CLARIE: Handling clarification requests in a dialogue system,” *Research on Language and Computation*, vol. 4, no. 2-3, pp. 259–288, Oct. 2006.

[4] T. P. Spexard, M. Hanheide, and G. Sagerer, “Human-oriented interaction with an anthropomorphic robot,” *IEEE Transactions on Robotics, Special Issue on Human-Robot Interaction*, December 2007, to appear.

[5] T. Spexard, S. Li, B. Wrede, J. Fritsch, G. Sagerer, O. Booij, Z. Zivkovic, B. Terwijn, and B. Kröse, “BIRON, where are you? - Enabling a robot to learn new places in a real home environment by integrating spoken dialog and visual localization,” in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Beijing, P.R. China: IEEE, October 2006, pp. 934–940.

[6] H. Ishiguro, T. Kanda, K. Kimoto, and T. Ishida, “A robot architecture based on situated modules,” in *Proc. EEE/RSJ International Conference on Intelligent Robots and Systems (IROS’99)*, 1999, pp. 1617–1623.

[7] T. Laengle, T. Lueth, E. Stopp, and G. Herzog, “Natural language access to intelligent robots: Explaining automatic error recovery,” in *Artificial Intelligence: Methodology, Systems, Applications*, A. M. Ramsay, Ed. Amsterdam: IOS, 1996, pp. 259–267.

[8] R. Ross, R. Collier, and G. M. P. O’Hare, “Demonstrating social error recovery with agentfactory,” in *AAMAS ’04: Proc. of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, Washington, DC, USA, 2004.

[9] R. Gockley, J. Forlizzi, and R. Simmons, “Natural person-following behavior for social robots,” in *Proc. Human-Robot Interaction*, 2007.

[10] S. Li, B. Wrede, and G. Sagerer, “A computational model of multi-modal grounding,” in *Proc. SIGdial workshop on discourse and dialog, in conjunction with COLING/ACL*. Sydney, Australia: ACL Press, July 2006.

[11] S. Li and B. Wrede, “Why and how to model multi-modal interaction for a mobile robot companion,” in *Proc. AAAI Spring Symposium on Interaction Challenges for Intelligent Assistants*, Stanford, 2007.