Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Why robots should be technical: Correcting mental models through technical architecture concepts

Lukas Hindemith, Jan Goepfert, Christiane Wiebel, Britta Wrede, Anna-Lisa Vollmer

2022

Preprint:

This is an accepted article published in Interaction Studies. The final authenticated version is available online at: https://doi.org/10.1075/is.20023.hin

Why robots should be technical

Correcting mental models through technical architecture concepts

Lukas Hindemith[1], Jan Philip Göpfert[1],
Christiane B. Wiebel-Herboth[2], Britta Wrede[3], and
Anna-Lisa Vollmer[3]
[1] CoR-Lab Bielefeld University | [2] Honda Research Institute Europe |
[3] Medical Assistance Systems Bielefeld University

Research in social robotics is commonly focused on designing robots that imitate human behavior. While this might increase a user's satisfaction and acceptance of robots at first glance, it does not automatically aid a non-expert user in naturally interacting with robots, and might hurt their ability to correctly anticipate a robot's capabilities. We argue that a faulty mental model, that the user has of the robot, is one of the main sources of confusion. In this work, we investigate how communicating technical concepts of robotic systems to users affect their mental models, and how this can increase the quality of human-robot interaction. We conducted an online study and investigated possible ways of improving users' mental models. Our results underline that communicating technical concepts can form an improved mental model. Consequently, we show the importance of consciously designing robots that express their capabilities and limitations.

Keywords: human-robot interaction, mental models, interaction design, robot architecture, technical concepts, robot transparency

1. Introduction

In recent years, the field of robotics has advanced rapidly. With vacuum cleaning robots or even humanoid robots such as Pepper,[1] this field is no longer limited to the industrial sector. Instead, new applications in private lives emerge. This change leads to new challenges for robotic engineers and researchers. Robot's

1. https://www.softbankrobotics.com/emea/en/pepper [accessed: 2021-03-11]

https://doi.org/10.1075/is.20023.hin
Interaction Studies 22:2 (2021), pp. 244–279. ISSN 1572-0373 | E-ISSN 1572-0381
© John Benjamins Publishing Company

working environments are no longer static and consistent as in factories, where their behavior can be pre-programmed. Hence, robots need continuous adaptation to novel and dynamic environments. Moreover, interaction partners are no longer experts, but naive users who want an intuitive interaction with robots.

Citizens of modern societies are regularly confronted with technology and are therefore used to several de facto standards regarding interfaces and the current state of the art. Nevertheless, most are not specialists in specific technological fields. Hence, neither require nor possess extensive knowledge about the inner workings of technical systems. We henceforth refer to them as *naive users.*

Intuitive human-robot interactions can be approached from contrary directions. Developing robotic systems simulating real humans' response and behavior is a common approach in current research, thus convincing users to have emotions (Breazeal et al., 2016; Vollmer & Schillingmann, 2018). While this approach reduces the cognitive load for users and at the same time increases the satisfaction on the user's side, it also raises problems (Breazeal et al., 2016; Duffy, 2006; Hassenzahl et al., 2020; Hegel et al., 2011). Today's robotic systems are still limited in their functionality and cannot cover the range of human capabilities. In particular, the way robots learn differs exceedingly from the way humans do (Vollmer et al., 2016). Thus, resulting in errors in human-robot interactions. In such situations, naive users are prone to be unable to trace the error back to its origin. This problem is due to the user's faulty *mental model* about the robot.

We understand a *mental model* as a cognitive framework people use to form an internal representation of the things they interact with (Staggers & Norcio, 1993). People build initial mental models of things they are unfamiliar with based on expectations and prior experiences. This initial mental model changes continuously based on new experiences. Thus, convincing naive users that social robots possess human internal mechanisms causes users to utilize knowledge about human-human interactions to build their mental models of robots. While this might work to some degree, a faulty mental model will cause incomprehensibility in error situations. Moreover, it causes erroneous user conduct, both eventually leading to misunderstandings and dysfunctional human-robot interaction. We consider learning interactions in particular as a relevant use case. When a user teaches new skills or knowledge to a robot a faulty mental model can lead to severe consequences. This might cause wrong sample generation or even learning incorrect skills, which takes both further away from a mutual mental model.

Based on this mismatch between the user's mental model and the actual functionality of the system, we argue social robots should not simulate biological functions and behavior. Instead, robots need to communicate concepts with their real functionality and limitations. However, this information should not overload the users. An improved mental model on the user's side will reduce the number of

erroneous human-robot interactions. With knowledge about the system's functioning and limitations, naive users will be more proficient in coping with errors.

We think that a new direction of research should focus on communicating insights into robots' architecture. To investigate this, we developed two complementary ways to communicate insights of a robotic system. These are prior instructions and a robot feedback system as visualizations. We evaluated their influence on the user's mental model in an online survey.

The remaining parts of the paper are structured as followed. In Section 2 we review factors that influence a mental model and ways to provide insights about robotic systems. Based on this, we formulate our hypotheses in Section 3. Section 4 includes our system realization and our study design. In Section 5 we present our results from the conducted online survey. Subsequently, we discuss the results and relate our findings to the hypotheses in Section 6. In Section 7 we summarize our work and provide an outlook to future work in Section 8.

2. Related work

2.1 Dual nature of computational artifacts: Relevance and architecture

An essential characteristic of computational artifacts (like robots) – similar to biological agents – is their dual nature: While an internal mechanism or algorithm generates a behavior, it can be observed from the outside (Rahwan et al., 2019; Schulte & Budde, 2018). Therefore, the field of didactics of computer science differentiates between the *relevance* of a computational artifact – which for the user is perceived as its function, e.g. the capability to autonomously drive in an environment or to execute spoken commands – and the *architecture* which is the algorithm or mechanism that produces this behavior e.g. the processing chain from perception over reasoning to action making use of abstractions such as object categories and states. It has been postulated that it is critical to make learners aware of the difference between relevance and architecture (Schulte & Budde, 2018). This is not a trivial task as humans tend to have intuitive mechanisms to predict other biological agents' behavior based on their own experiences.

2.2 Relation to human interactive learning and pragmatic frames

When teaching children, humans exhibit a range of highly adaptive behaviors that tailor learning input to the learner's capabilities and understanding and facilitate learning by directing attention and structuring the interaction (Brand et al., 2002; Nelson et al., 1989; Pitsch et al., 2014; Vollmer et al., 2009). One essential

strategy utilized by parents is the use of *Pragmatic Frames,* recurring interaction patterns that allow the learner to use experiences from known, previous interactions in novel situations. This supports the process of abstracting from context (Bruner, 1985; Rohlfing et al., 2016). The potential of pragmatic frames for human-robot interaction has been described by Vollmer et al. (2016). When teaching a robot, humans seem to intuitively make use of pragmatic frames (Hindemith et al., 2019). However, while using such strategies is beneficial for children, they may not be for current robot systems. For example, some learning algorithms rely on randomly sequenced learning data whereas teaching in context is based on repeating and slightly modifying an action, again and again, consequently leading to clusters of similar data in the input. Thus, these strategies are well-tuned to the human mind. This includes the ability for the theory of mind (TOM), i.e., to take the perspective of another person. This requires a mental model of the other person consisting of her physical capabilities such as perception and action alongside her mental states like goals and intentions and even emotional states (Sterelny, 1990). Humans tend to apply such a model to technical artifacts as well, as the involved processes are highly intuitive. However, as the cognitive architecture of technological artifacts, such as robots are very different from a biological human mind, this often leads to misunderstandings and failed interactions. In this work, we investigate how far humans can benefit from technological concepts for the formation of correct mental models while interacting with robots.

2.3 Communicating technical concepts

We hypothesize the communication of technical concepts can be helpful to improve users' mental models about robots. Information communication for human-robot interaction is usually realized via instructions that should be as intuitive and accessible to non-experts as possible. Another strand of research communicates information directly via implicit or explicit robot feedback during the interaction.

2.3.1 *Instructions*

While experts are experienced in operating their system, without further information, naive users fail to do so. Therefore, it is necessary to provide new users with supplemental materials to increase system understanding. This additional information can be provided in various ways. For example, information can be provided as a manual to read, a video to watch, or a tutorial where users also take action. Cakmak and Takayama (2014) investigated the influence of these three channels on the users' ability to successfully interact with a robot. The different types of instructional material were evaluated in a *Programming by Demonstra-*

tion scenario. As a result, interaction videos mediate the needed knowledge the best. Based on these results, we also utilize interaction videos in our study.

2.3.2 *Feedback*

Feedback is an essential factor in communicating the unobservable processes of the robot. This, in turn, influences how users respond to interactions with a robot. Communication of inner processes to the user can improve the comprehension of the decision making process of the robot (Wortham et al., 2017). Users prefer a robot that provides feedback, even though it might malfunction at some point (Hamacher et al., 2016).

Concerning *how* feedback should be provided, Breazeal et al. (2005) showed that a combination of explicit and implicit feedback improves the effectiveness of human-robot interaction because the malfunction of the robot can be more easily detected and recovered, in contrast to only explicit feedback. Thomaz and Cakmak (2009) examined the learning performance of a robot that learned objects in a social way and a non-social way. In the social condition, the robot used gaze behavior to indicate errors, which besides better sampling from human partners led to faster error recovery in the interaction. While this paper focused on implicit feedback, Otero et al. (2008) examined the impact of explicit verbal feedback on the perception of user demonstrations. The majority of the subjects repeated the same demonstration until positive feedback was given by the robot, instead of trying to optimize the given demonstration. Other approaches on robot feedback aim to communicate what the robot learned (e.g., de Greeff and Belpaeme, 2015; Vollmer et al., 2014) and their execution capabilities (e.g., Kwon et al., 2018). In our work we used an introduction as an explicit way, and the visualization as an implicit way to communicate the inner working of the robot.

3. Hypotheses

Based on the goal to improve human-robot interactions by shaping an appropriate mental model of the robot, we hypothesize the following:

Hypothesis 1: Providing architectural concepts allows users to gain more knowledge about the functionality of a robot.

Hypothesis 2: Insights into the architecture of a robot increases the ability to recognize and explain errors in human-robot interaction.

Hypothesis 3: Technical concepts differ in terms of their familiarity and observability. These factors influence the user's ability to recognize and understand problems in human-robot interactions.

4. Methods

To investigate our hypotheses, we developed two ways of providing architecture insights about the robot: (a) architecture information is given in an *instruction video* before the human-robot interaction and (b) a *visualization* of the current internal states of the robot is shown to users along with the interaction with the robot.

We conducted an online study devised as a survey. In this survey, participants watched different erroneous human-robot interaction videos and answered questions regarding the source of the underlying problems.

4.1 Scenario

For the scenario, we developed our system with regard to an object learning interaction. The goal of this scenario was to teach the robot a label for an object. The object recognition was realized by using Aruco marker detection (Garrido-Jurado et al., 2014). These markers were attached to the objects to uniquely identify each object. The learning of a label for an object was realized by storing a map between an Aruco marker ID and the verbally provided label. The scenario was selected to incorporate established concepts in human-robot interactions. In the following, we will give you a more in-depth view of the used system and the mentioned concepts.

4.2 System and concepts

Our approach was realized on a robotic system. The technical concepts were selected based on the setup of the robot and the implemented scenario.

4.2.1 *Robot*

Robotic platform

We used the robotic platform *scitos G5* by Metralabs.[2] The robot is equipped with two RGB-D cameras. One is mounted on a pan-tilt unit at the top of the robot. The other one in front of the robot. To display information, a touch display is mounted above the second camera. Behind the display, a microphone and two speakers are located. The robot is also equipped with a 6 DOF robotic arm and a mobile base with front and rear lasers, which were unused in this study.

2. https://www.metralabs.com/en/ [accessed: 2021-03-11]

System
The robot was developed based on ROS (Stanford Artificial Intelligence Laboratory et al., 2014). The robot's main functionalities for the scenario were:

- object recognition
- speech recognition
- speech synthesis
- behavior control

The *object recognition* was developed using the Aruco marker detection to overcome the problem of unstable object recognition. The RGB images from both cameras were used to detect these markers in the world. Objects were equipped with a marker, which had a unique identifier. Thereby the system was able to track each object, even if it was not visible for the robot for a certain amount of time. For the *speech recognition,* we used the google service *Cloud Speech-to-Text*.[3] To reduce the amount of recorded audio by google, we used an additional wake-up-word detector. To interpret the recognized speech, we developed a grammar parser based on the Backus-Naur-Form (McCracken & Reilly, 2003). The *speech synthesis* was realized by a voice synthesizer. For the behavior control, we developed a finite state machine, based on the *flexbe* engine (Schillinger et al., 2016).

4.2.2 *Concepts*

Based on the developed robotic system, we selected the three most common concepts in current robotic systems: object recognition, speech recognition, and finite state machine. Furthermore, these concepts have been operationalized in error situations presented in videos.

Object recognition
The used Aruco marker detection is similar to the scanning of QR codes (Soon, 2008) or bar codes (Beller & Wang, 1997), which is used in various real-life scenarios. This allows us to investigate the **third hypothesis,** which states that familiarity and observability influence the ability to recognize and explain errors. Therefore, we did not use an object recognition system that detects the actual objects but recognizes objects by their corresponding Aruco marker.

Speech recognition
While our speech recognition software was able to recognize natural language, it was unable to infer the intent of the commands. Therefore, we used a grammar parser to assign intent to the speech command. Because the concept of having

3. https://cloud.google.com/speech-to-text [accessed: 2021-03-11]

limited speech understanding capabilities is important for users to know, we communicated this concept.

State machine
A frequently used way to control the robot's behavior is a finite state machine. In state machines, the robot's actions are realized as states. Each state provides an outcome that can lead to other states. By designing such a set of states and connections between them, the robot can fulfill a predefined goal. The higher the flexibility of a robot's behavior should be, the more work it costs to design a corresponding state machine. Consequently, state machines are usually limited in their flexibility and therefore the user has to stick to the pre-programmed sequence of actions for the robot to work properly. For the user to be able to follow the correct sequence, knowledge about the state machine and its sequence has to be provided.

4.3 Experimental design

The design decisions to communicate the architecture information were based on the concepts mentioned above. For providing such insights about the robot, we decided on two complementary approaches. The first approach was to design an instruction video that gives direct explanations of the robot's architecture and functionality. This video is displayed to one group of users before the interaction with the robot. The second approach communicates these concepts indirectly by providing visualizations about the internal states of the robot. This visualization is shown to users while they are interacting with the robot. In the following, we will discuss each approach in more detail.

4.3.1 *Architecture instruction video*

The instruction video was designed to communicate the technical concepts of the robot in a direct way. Our goal was to inform about the software and hardware features of the robot equally while avoiding the cognitive overload of the user. Hence, the video combined information about the hardware components and their functionality on the software level. To reduce possible confusion of the user, each concept was introduced independently from the other concepts. To direct the user's attention only to one aspect at a time, each fact about a concept was printed in a box and shown one after another. Previous boxes that correspond to the same concept were still displayed. For an improved understanding of the used hardware, a 3-dimensional model of the robot was displayed in the center of the video. The corresponding hardware of each concept was also highlighted by colored ellipses as an overlay of the 3-dimensional model. An arrow between the text box and the colored ellipses helped to connect the information. To improve

the understanding of the displayed information, the text boxes are also read out. Thereby the user is stimulated in a multi-modal way, which improves the information reception (Sweller et al., 2019).

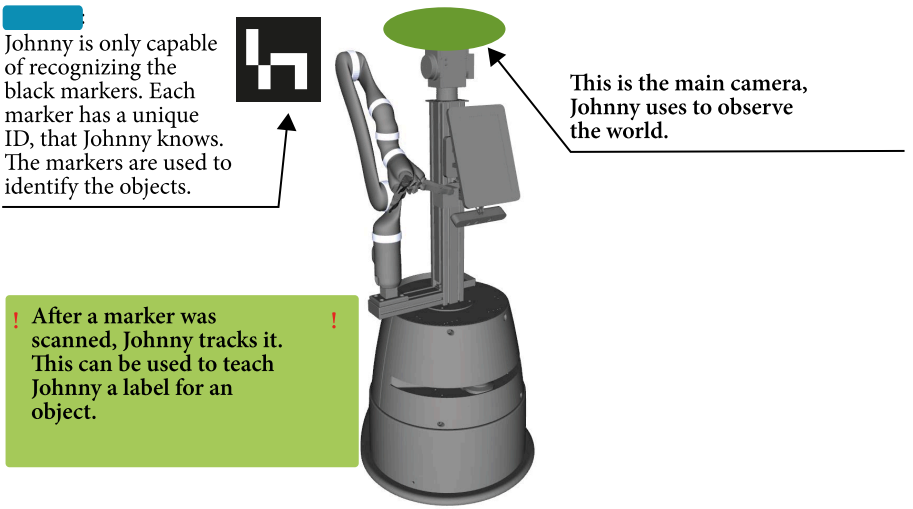

Figure 1. Sample scene from the instruction video that describes the camera of the robot. The text boxes were shown in the order: Top right, top left, bottom left. The text in the top right describes the camera as a hardware feature. The text in the top left informs about the marker detection. How this marker detection can be used is described in the bottom left information box.

The focus of the insights was to mediate the general concepts, their possibilities but also limitations. No details about the underlying algorithms or their implementations were given. Hence, the information text was designed to contain technical terms while still be simple enough to be understandable for naive users. In addition to this, the wording should not convey any analogies to human characteristics. For example, wording such as "*With the microphone, the robot hears the user*" for the speech recognition module would imply the robot can hear like a human. Instead, wording such as "*The microphone […] is used to process speech input from the user.*" emphasizes the technicality of the system. The text boxes of a concept were arranged to first mention the used hardware, followed by the software side usage. As a follow-up, additional notes on how this information can be used were mentioned in a green highlighted box. Refer to Figure 1 for an example frame from the introduction video, in which the robot's camera is introduced together with object recognition.

4.3.2 *Robot visualization*

While the instruction video was shown before the interaction, the visualization of the internal states of the robot was shown alongside the interaction. To not disturb the interaction, the design communicated the technical concepts of the robot more indirectly. Consequently, this approach leaves more room for interpretation in contrast to the instruction video, as the visual elements were not explained. In the following, our visualizations of these concepts are described (cf. Figure 2).

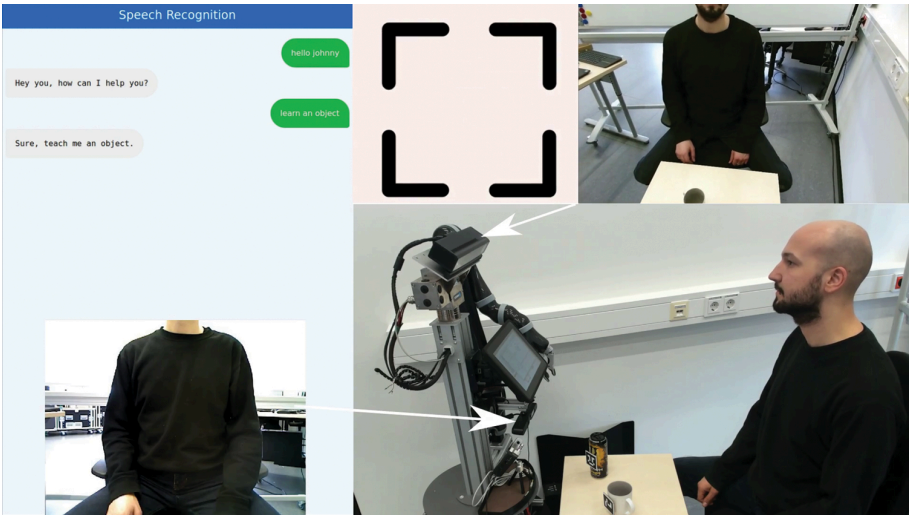

Figure 2. Frame of the interaction video with additional visualizations, describing the internal states of the robot. Visual elements from top left to bottom right: Speech recognition, state machine, main camera stream, scanning camera stream, human-robot interaction.

Marker detection for object identification

To communicate the visual perception of the robot, we showed the current image stream from the head camera of the robot. The detected Aruco markers were visualized as overlays of the image stream. Therefore, the overlays are bounding boxes around the detected markers. The bounding boxes are colored depending on the current status of the marker. The status of a marker can be *focused* or *not focused*. A focused marker indicates the current interaction is centered around this marker. If a marker is focused on by the robot, the color of the associated bounding box is colored green. Otherwise, the bounding box is colored red. The design choice of the colors is the same as for the speech recognition (cf. Section 4.3.2). Stored

labels for Aruco markers were printed above the corresponding bounding box and were colored the same (cf. Figure 3).

Figure 3. Clipped sample image of the object recognition visualization. The left marker is not tracked and therefore red. The right marker is currently tracked by the robot, which is indicated by a green bounding box. The learned label by the robot of each marker is written above the bounding boxes.

Verbal communication

The main communication between the robot and the user was verbal. To trace the history of the interaction, the visualization contained the utterances of both the robot and the user. Because the process of speech synthesis by the robot did not influence how the user is expected to behave, no further information was provided about this concept.

The speech recognition process was visualized in a way that several parts of the communication were displayed. As the robot was not constantly listening to speech input, the user should at all times be able to notice whether the robot is listening to speech input. After an utterance was recognized, a visualization should show *what* was recognized and *whether* the command could be interpreted by the

robot. With these types of information, the user is more likely to be able to detect potential errors in the speech recognition process.

The overall layout was designed with the aim of intuitiveness and familiarity. Accordingly, we decided to visualize the verbal communication in the style of a chat box, such as WhatsApp[4] or Messenger.[5] With 2×10^9 monthly users for WhatsApp and 1.3×10^9 monthly users for Messenger in October 2019 (Clement, 2020) the design of chat boxes is familiar to a significant percentage of the population.

Our design, with exemplary input, can be seen in Figure 4. The arrangement of the text boxes was based on the perspective of the user. Therefore, the speech output of the robot was printed in a speech bubble on the left side of the chat box. The right side shows the speech recognition of the command provided by the user. This allocation of perspective is the de facto standard in messengers with such a design. To communicate when the robot listened to speech input, a blue speech bubble with three dots was displayed. This visualization is used by the WhatsApp messenger to communicate that someone is currently writing a message. After the speech input was processed by the robot, the resulting recognition was displayed. At this point, we differentiated whether an intend could be determined or not. If the robot could determine an intend, the speech bubble of the corresponding speech input had a green background color. Otherwise, the background color of the speech bubble was red. The color decision was based on the *positive* respectively *negative* meaning of the green and red colors in our modern society. For example, traffic lights use the color green to communicate that a driver is allowed to drive, respectively red to communicate the driver must wait. Additionally, the color red mediates possible danger.

Finite state machines for robot control

Based on the highly mathematical nature of finite state machines, we expected this technical concept to be incomprehensible to most naive users. Therefore, our goal was to visualize a simplification of this process. Instead of visualizing the entire sequence, we only visualized the current state. To make the current state comprehensible, while requiring as little attention as possible, each state was represented as an icon. Figure 5 shows the five icons used in our visualization to communicate the most important states of the state machine.

The designs for the icons were made by *kiranshastry* from *www.flaticon.com*.[6] Our approach was to use simple icons, follow a uniform design among each other,

4. https://www.whatsapp.com/ [accessed: 2021-03-11]
5. https://www.messenger.com/ [accessed: 2021-03-11]
6. https://www.flaticon.com/authors/kiranshastry [accessed: 2021-03-11]

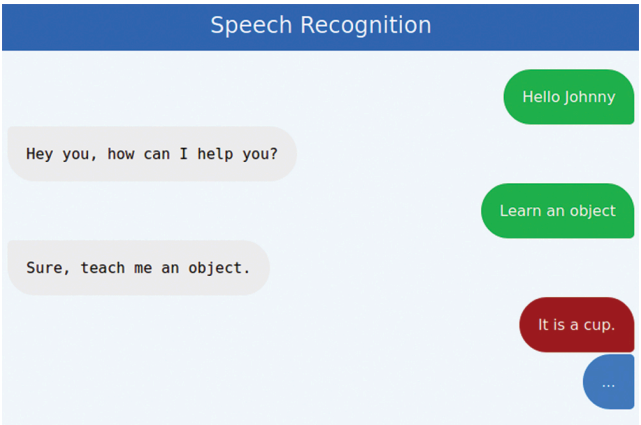

Figure 4. The chat box visualization for the verbal communication. The robot's speech output is on the left. The right side displays the speech recognition of the robot. Whether the robot currently listens to speech is indicated by a blue speech bubble with three dots. The recognized speech is written in a speech bubble on the right side as well. If the speech could be parsed by the robot, it is displayed in a green speech bubble (e.g. first and second recognized speech). Otherwise it is written in a red speech bubble (e.g. last recognized speech).

Figure 5. The used icons to visualize the current state of the robot. The icons from left to right: The robot is currently speaking (`robot speaks`), the robot listens for speech input (`robot listens`), the robot scans the room for a marker (`robot scans`), the robot stores new information in its memory (`robot stores`), the robot is currently occupied with a task (`robot works`).

and are intuitive. Based on these guidelines, the symbols for speech recognition (`robot listens`) and speech synthesis (`robot speaks`) are based on symbols used by Microsoft's operating system Windows, which is the most popular operating system worldwide (Liu, 2020). Since QR markers are used in many areas of application, the scanning symbol (`robot scans`) is expected to be familiar to many people. The symbol to store new information (`robot stores`) is a combination of a document and an *add* (plus) symbol. This combination of two familiar icons facilitates an intuitive understanding. The last symbol is a robot and is used whenever the robot is doing something where the user needs to wait (`robot works`). While this

design might not be known from somewhere else, a robot face can be easily identified while still be very simple in its design.

4.3.3 *Course of online study*

To verify our hypotheses we conducted an online survey with 130 participants. In this survey, participants had to detect and further elaborate on erroneous human-robot interactions in videos. Furthermore, parts of the participants' mental model about the robot were measured. We divided the participants into four different condition groups. Each group was shown a different amount of information about the robot. To measure the impact of providing technical concepts of the robot to the participants, we used a 2×2 study design. The baseline was only shown the interaction video. The `Instr` and `Vis` groups received additional information based on one of the approaches each. The fourth group `Instr+Vis` received the combined information of both approaches (cf. Table 1).

Table 1. The condition groups of the online survey and their amount of provided information

Condition type	No instruction	Instruction
No Visualization	Baseline	Instr
Visualization	Vis	Instr+Vis

The online survey was carried out using the online portal *Prolific*[7] to acquire participants. The course of the survey is illustrated in Figure 6. First, the participants were welcomed and asked general questions about their person. Furthermore, we measured their technical affinity, using the *Affinity for Technology Interaction Scale* (Franke et al., 2019a).

After the general questionnaires, each participant was assigned randomly to one of the four condition groups. Depending on the condition to which the participants were assigned, they were shown an instructional video with architectural information about the robot or not (cf. Section 4.3.1). Afterward, a video of a successful human-robot interaction, following the scenario, was presented. Depending on the condition, the interaction video was enriched with information about the internal states of the robot (cf. Section 4.3.2). To check the participants' knowledge about the robot's features, open questions were asked. We separated features of the robot into *hardware* and *software* (cf. Table 2). Afterward, three erroneous human-robot interaction videos were presented to the participants in a randomized order. While the interaction in the videos stayed the same for all conditions, the `Vis` and `Instr+Vis` conditions were shown enriched videos in the same way as

7. https://www.prolific.co [accessed: 2021-03-11]

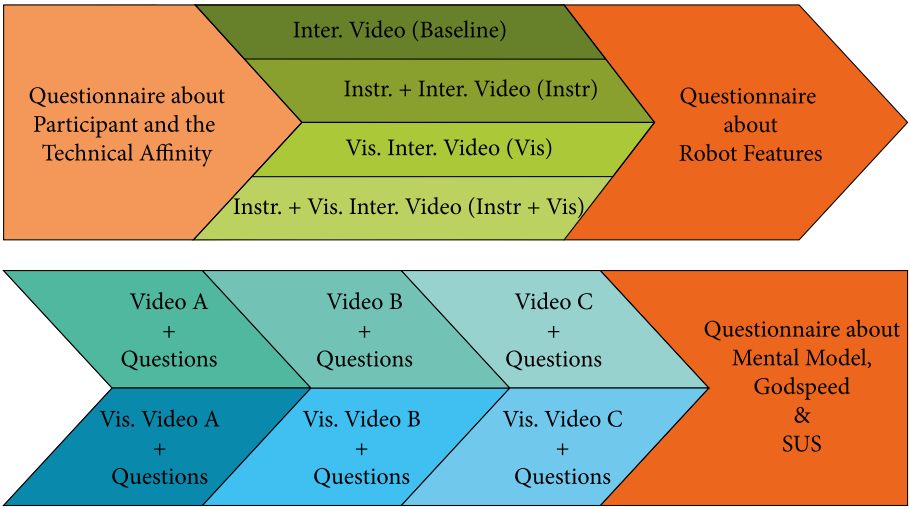

Figure 6. Course of online study, starting from top left to bottom right. At the beginning, participants were asked to fill out questionnaires about themselves and their technical affinity. Afterwards, depending on the condition, the participants were shown an instruction video and an interaction video. In a follow-up questionnaire about the robot features, the participants' knowledge about the robot was surveyed. Afterwards, the erroneous interaction videos were shown and related questions were asked. In the end, questionnaires about the mental model, *godspeed* and *SUS* were surveyed.

the initial interaction video. After each video, several questions about the origin of the error and how participants observed it were asked (cf. Table 2). Subsequently, each participant was asked open questions about the technical concepts that led to the errors and what they associate with them. We also collected parts of the *Godspeed* (Bartneck et al., 2009) and *SUS* (Bangor et al., 2008) questionnaires in order to measure subjective ratings with regard to the robots' appearance.

Table 2. The open questions asked throughout the survey

Category	Question
Hardware features	Which components does the robot have that allow it to observe or interact with its environment.
Software features	What skills and abilities does the robot have?
Interaction problem	What went wrong during the interaction you saw in the last video?
Origin of error	Why did the problem occur?
Error observation	How did you recognize the mistake?
Error correction	What would have been correct?

4.3.4 *Human-robot interaction videos*

For the human-robot interaction, we have implemented an object labeling scenario (cf. Section 4.1). The general interaction flow is shown in Figure 7. First, the human greets the robot, upon which the robot welcomes the human and asks for a task. The human triggers the object learning interaction by a command, which the robot confirms. To teach a label for an object, first, the human shows the object of interest into the scanning camera of the robot. Afterward, the human provides the label of the object verbally. The robot confirms the learning and ends the interaction. For the erroneous interaction videos, the interaction had violations at some point each. But in each video, there is no more than one error. The errors were based on the concepts described in Section 4.2.2 and induced by the human.

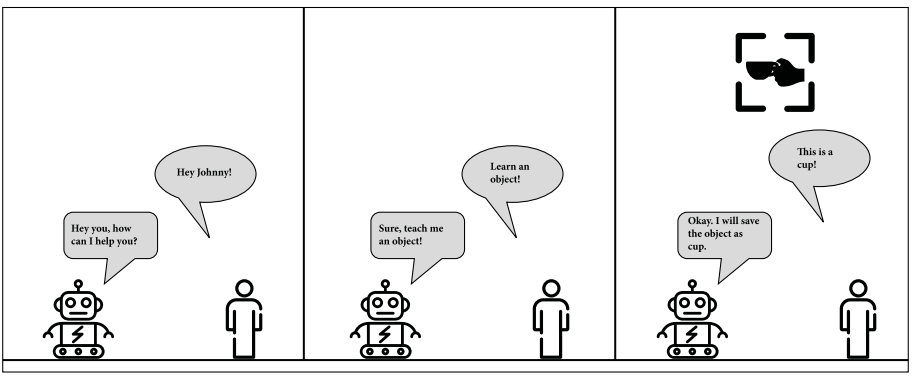

Figure 7. The interaction flow of the object labeling scenario, from left to right.

Object detection error

One of our technical concepts was the Aruco marker detection as a way to identify objects. To generate an error that aims at this concept, the user fails at pointing the marker towards the robot and instead rotates the marker sideways in front of the external camera. Because the robot cannot detect any marker, it encounters a timeout, which causes the robot to switch to a "failed" mode, where it says that something went wrong, then ends the interaction. In the following, we refer to this video with `No Object`.

Speech recognition error

Because the robot can only process exact commands, we induced an error by rephrasing the utterance to provide the label. Therefore, instead of saying *"This is a cup"*, the user says *"It is a cup"*. Both sentences have the same meaning while the wording differs. Because the robot cannot process the command, it switches to the

mentioned "failed" mode, then ends the interaction. In the following, we refer to this video with `Failed Speech`.

State machine error
For the concept of state machines, the sequence of actions was violated at some point. After the robot switches to the "learning" mode, the robot expects to scan an object. Instead, the user first provides the label and tries to show the object afterward. Consequently, the robot switches to the "failed" mode after receiving the label and ends the interaction. In the following, we refer to this video with `Failed SM`.

5. Results

The analyses were based on 122 out of 130 participants. 8 participants were excluded due to incomplete entries. All participants were located in the United States (63 women, 56 men, 2 diverse, 1 anonymous, $M_{age} = 34.09$ years, age range: 18–74). Each participant was assigned randomly to one of the four condition groups (24 `Baseline`, 38 `Instr`, 29 `Vis` and 31 `Instr+Vis`). Due to the randomization process of the survey software, condition Baseline has slightly less while condition Instr has slightly more participants than the other groups. Shapiro-Wilk normality checks (Shapiro & Wilk, 1965) showed non-normal distributed data, and follow up Kruskal-Wallis tests (Breslow, 1970) for gender and age did not indicate any significant differences between the condition groups. Hence, we had balanced condition groups in terms of participants.

5.1 Hypothesis 1: *Providing architectural concepts allows users to gain more knowledge about the functionality of a robot*

To investigate this hypothesis, each participant was asked questions regarding the *hardware* and *software* features of the robot after the introduction. To analyze the answers to the open questions, we assigned each one of them the features mentioned. We only focused on features that were important for the human-robot interaction in the videos. As key figures, we calculated the number of features each participant mentioned. In a second step, we took a detailed look at each feature and how many participants mentioned them.

Which components does the robot have that allow it to observe or interact with its environment? (hardware)

The important features for the hardware of the robot were the ***camera,*** the *microphone,* the *pan-tilt unit (PTU),* and the *speaker.* To observe the influence of the instruction video on the knowledge about the robot, the *speaker* was not mentioned in the introduction video but was included in the analyses.

We first applied a Shapiro-Wilk normality check, which indicated a non-normal distribution of the data. Therefore, we used the Kruskal-Wallis test, which showed a significant difference in the number of mentioned hardware features. A follow-up post-hoc Dunn test (Dunn, 1964) revealed that the conditions `Instr` and `Instr+Vis` mentioned significantly more than the `Baseline` (0.0403 and 0.0163) and the `Vis` (0.0220 and 0.0081) conditions (cf. Table 3).

Table 3. Shapiro-Wilk normality check and Kruskal-Wallis, with follow up post-hoc Dunn test for the number of hardware features mentioned between all conditions, p-values < 0.05 are highlighted in **bold**, **Shapiro Wilk** (statistic = 0.8787, p-value = **1.5–08**), **Kruskal-Wallis** (H-statistic = 11.20, p-value = **0.01**)

Condition 1	Condition 2	p-value
Instr	Baseline	0.0403
	Vis	0.0220
	Instr+Vis	0.6243
Vis	Baseline	0.9127
	Instr+Vis	0.0081
Instr+Vis	Baseline	0.0163

A closer look at each hardware feature (cf. Figure 9) turned out that the *microphone* and *PTU* features had differences in their frequency (Kruskal-Wallis test). The follow-up post-hoc Dunn test (Dunn, 1964) revealed that the conditions `Instr` and `Instr+Vis` mentioned the *microphone* feature significantly more frequently than the `Baseline` and `Vis` conditions. Furthermore, the Instr condition mentioned the *PTU* features more often than the `Baseline` and `Vis` conditions (cf. Table 4). The *camera* and *speaker* features had no differences in frequency between the conditions. In general, most participants mentioned the *camera* feature, in contrast to the other features.

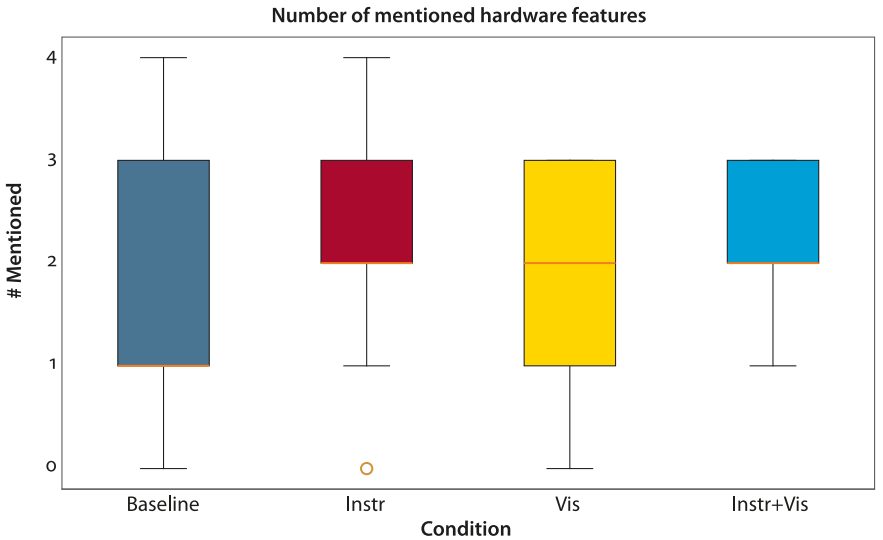

Figure 8. Number of mentioned hardware features per participant

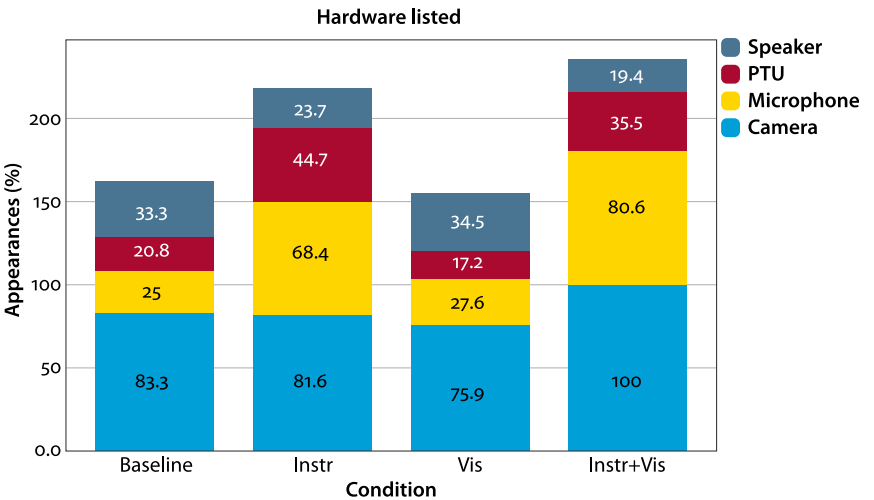

Figure 9. Frequency of each hardware feature

Table 4. Kruskal-Wallis test, with follow-up post-hoc Dunn test for the frequency of mentioned hardware features. Below the features are the p-values of the Kruskal-Wallis test. Entries are p-values of the Dunn test. p-values < 0.05 are highlighted in **bold**, p-values < 0.1 are highlighted in *italic*

		Hardware feature			
		Camera	Microphone	PTU	Speaker
Condition 1	**Condition 2**	**0.5317**	**0.0000**	***0.0701***	**0.1971**
Instr	Baseline	–	**0.0001**	**0.0303**	–
	Vis	–	**0.0009**	**0.0227**	–
	Instr+Vis	–	0.9609	0.1462	–
Vis	Baseline	–	0.6017	0.9413	–
	Instr+Vis	–	**0.0012**	0.3423	–
Instr+Vis	Baseline	–	**0.0002**	0.3913	–

What skills and abilities does the robot have? (software)

As we did for the hardware features, likewise we did for the software features. Again, we only took into account the features relevant to the interaction videos. These were:

- object recognition
- speech recognition
- speech synthesis
- learning

Figure 10 illustrates how many software features each participant mentioned. A Shapiro-Wilk normality check showed a non-normal distribution, and the Kruskal-Wallis test indicated no significant differences between the conditions (p-value = 0.1092).

We also analyzed how often each software feature was mentioned by the condition groups. We then applied the Kruskal-Wallis test with a subsequent Dunn test. As shown in Figure 11, all conditions mentioned the software features almost equally often. Tests showed only for the *speech synthesis* that the Vis condition mentioned this feature significantly more frequently than the Instr condition (p-value = 0.0167) and a trend in the difference for the Instr+Vis condition (p-value = 0.0701) (cf. Table 5).

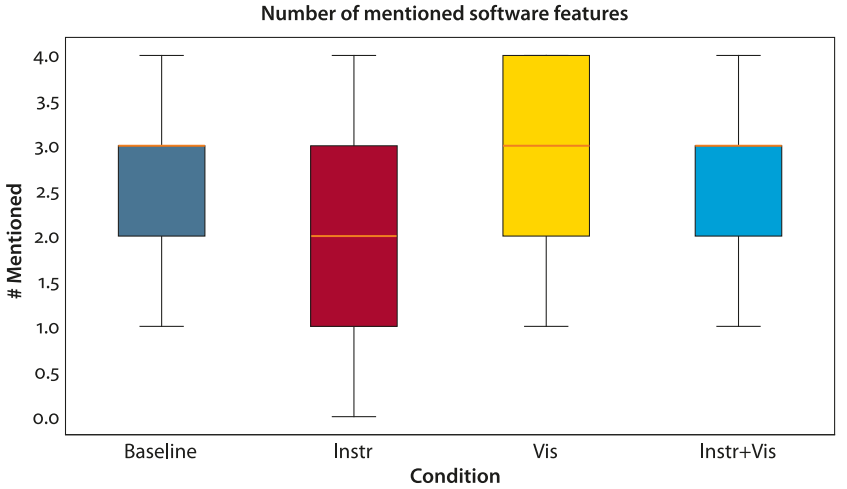

Figure 10. Number of mentioned software features per participant

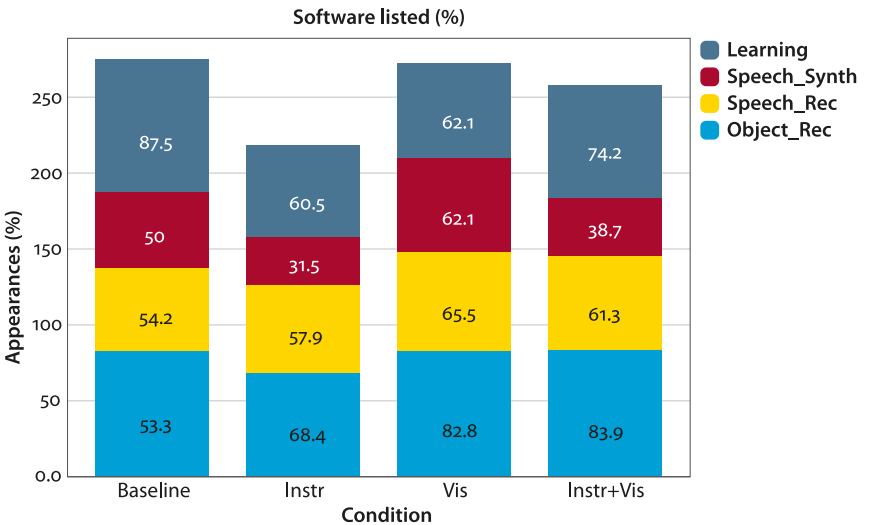

Figure 11. Frequency of each software feature

Table 5. Kruskal-Wallis, with follow-up Dunn test for the frequency of mentioned software features. Below the features are the p-values of the Kruskal-Wallis test. Entries are p-values of the Dunn test. p-values < 0.05 are highlighted in **bold,** p-values < 0.1 are highlighted in *italic*

		Software feature			
		Object Rec.	Speech Rec.	Speech synthesis	Learning
Condition 1	**Condition 2**	0.4465	0.8674	*0.0912*	0.1279
Instr	Baseline	–	–	0.1793	–
	Vis	–	–	0.0167	–
	Instr+Vis	–	–	0.6055	–
Vis	Baseline	–	–	0.3809	–
	Instr+Vis	–	–	*0.0701*	–
Instr+Vis	Baseline	–	–	0.4055	–

Concerning the hypothesis, the results showed that participants who received architectural information in the form of an instruction video could name more hardware features of the robot. The same effect was not visible for the software features of the robot.

5.2 Hypothesis 2: *Insights into the architecture of a robot increases the ability to recognize and explain errors in human-robot interaction*

To examine this hypothesis, we analyzed the answers to the open questions after each video (cf. Table 2). First, we checked if the participants recognized the error and could explain why the error occurred. We chose a binary format to rate the correctness of each answer. The rating was based on the factors cause and effect of the error. The first analysis showed that if a participant could answer *what* happened, the question regarding *why* this error occurred was answered correctly as well. Due to this, and some participants answered the *what* question too general, we only focused on the explanations *why* the error occurred, for further investigations.

Figure 12 shows how many participants provided correct explanations for each interaction video. The Kruskal-Wallis test indicated a significant difference between conditions for each video and all videos combined (cf. Table 6). While the post-hoc Dunn test already shows the most significant differences for the Instr+Vis conditions for each video, it surpasses all other conditions significantly in terms of the number of correct explanations for the combined results of all videos.

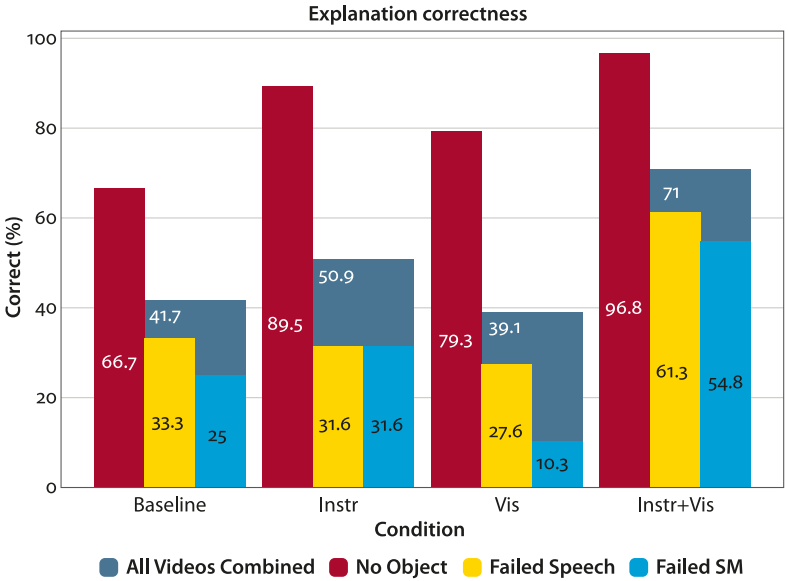

Figure 12. Correctness of the given explanations

Table 6. Kruskal-Wallis, with follow-up Dunn test for the number of correct explanations. Below the videos are the p-values of the Kruskal-Wallis test. Entries are p-values of the Dunn test. p-values < 0.05 are highlighted in **bold**, p-values < 0.1 are highlighted in *italic*

		Erroneous interaction			
		No Object	Failed Speech	Failed SM	Combined
Condition 1	**Condition 2**	**0.0142**	**0.0265**	**0.0026**	**0.0000**
Instr	Baseline	0.0163	0.8905	0.5874	0.2215
	Vis	0.2576	0.7404	*0.0640*	*0.0978*
	Instr+Vis	0.4074	0.0120	0.0388	0.0041
Vis	Baseline	0.2082	0.6610	0.2534	0.7457
	Instr+Vis	*0.0634*	0.0076	0.0002	0.0000
Instr+Vis	Baseline	0.0024	0.0354	0.0183	0.0002

These results show that our hypothesis is confirmed when combining both approaches for providing insights into the robot's architecture, but has to be rejected for each approach individually.

5.3 Hypothesis 3: *Technical concepts differ in terms of their familiarity and observability. These factors influence the user's ability to recognize and understand problems in human-robot interactions*

A key figure for this hypothesis was the number of correct explanations for each video (cf. Figure 13). The Kruskal-Wallis test indicated a significant difference between the videos (H-statistic = 81.1717, p-value = 0.000). The subsequent post-hoc Dunn test revealed that the `No Object` error was detected significantly more than the other errors (cf. Table 7). While the `Failed SM` error was noticed slightly less than the `Failed Speech` error, there was no significant difference.

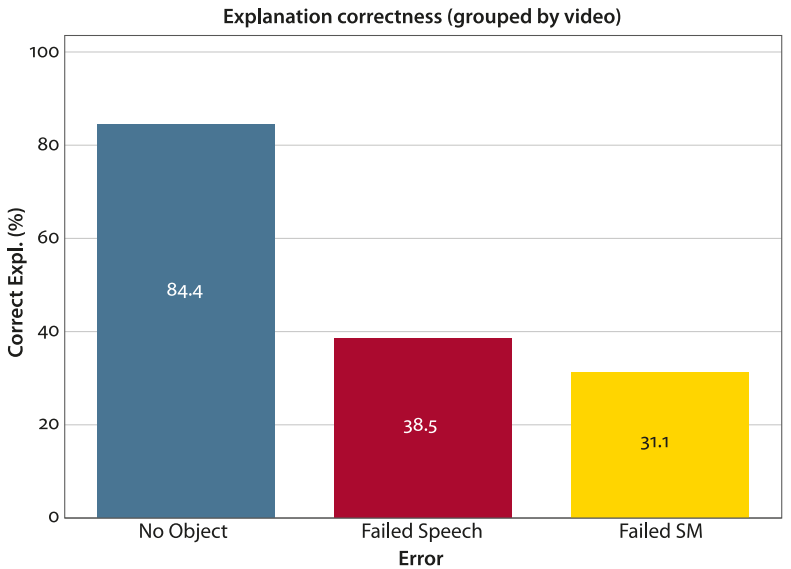

Figure 13. Amount of correct explanations per video

Table 7. Kruskal-Wallis, with follow-up Dunn test for the number of correct explanations per video. p-values < 0.05 are highlighted in **bold**, p-values < 0.1 are highlighted in *italic*

Condition1	Condition2	p-value
`No Object`	`Failed Speech`	**0.0000**
	`Failed SM`	**0.0000**
`Failed Speech`	`Failed SM`	0.2497

Kruskal-Wallis (H-statistics = 81.1717, p-value = **0.0000**)

Concerning the third hypothesis, we also asked the participants, after each video, how they recognized the error. To assign each answer to a category, we chose the following categories:

- `Instruction`: The initial instruction video
- `Initial Video`: The initial working Human-Robot-Interaction video
- `Interaction`: The current Human-Robot-Interaction video
- `Visualization`: The additional visualization about the robot's internal states
- `Other`: Could not be categorized to one of the others

The answers can be seen in Figure 14. A Kruskal-Wallis test indicated significant differences for the `Initial Video` and the `Interaction` category between the videos. A post-hoc Dunn test revealed that the `Failed Speech` and `Failed SM` error videos were significantly more often detected by the `Initial Video` than the `No Object` error video. In contrast to this, the `No Object` error video was detected significantly more often in the current interaction video than the other two videos (cf. Table 8). Based on these results, we can see that concepts differ in terms of familiarity and observability.

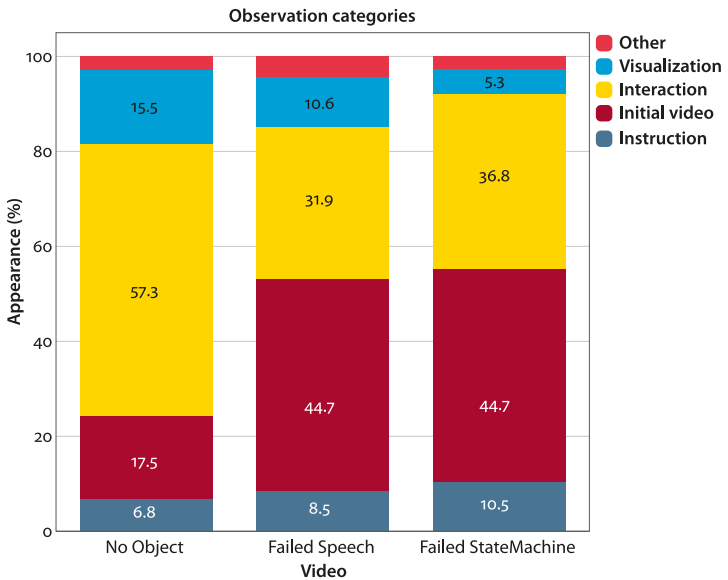

Figure 14. Categories of how the error was detected for correctly named error sources

Table 8. Kruskal-Wallis, with follow-up Dunn test for the frequency of error observation categories. Below the categories are the p-values of the Kruskal-Wallis test. Entries are p-values of the Dunn test. p-values < 0.05 are highlighted in **bold,** p-values < 0.1 are highlighted in *italic*

		Error observation	
		Initial video	Interaction
Condition 1	**Condition 2**	**0.0003**	**0.0061**
No Object	Failed Speech	0.0008	0.004
	Failed SM	0.0017	0.0314
Failed Speech	Failed SM	0.9955	0.6517

5.4 ATI, godspeed and system-usability-scale

At the beginning of the survey, participants were asked to fill out the *ATI* questionnaire (Franke et al., 2019b). Furthermore, the participants were asked to fill out parts of the *godspeed* and the *system-usability-scale (SUS)* questionnaire at the end of the survey. We only included the key figures *anthropomorphism, likeability* and *perceived intelligence* from the *godspeed* questionnaire. Other parts of the godspeed questionnaire focus on key figures that were unimportant for our analysis.

To evaluate the influence of a better understanding of the robot towards the questionnaire scores, we grouped all participants by the number of correctly explained erroneous videos (i.e., participants with zero correct explanations were in group 0, those who could detect one in 1 etc.). Shapiro-Wilk normality checks showed non-normal distributions for the key-figures *anthropomorphism, Likeability* and *SUS*.

The Kruskal-Wallis test showed significant differences for the *anthropomorphism* and *SUS* score. The follow-up post-hoc Dunn test revealed that participants who were able to explain all three errors had a significantly lower *anthropomorphism* score than those who detected fewer errors. In contrast, the *SUS* score was higher for participants who detected more errors (cf. Table 9). However, the *ATI* score, which reflects the technical affinity, did not influence the results.

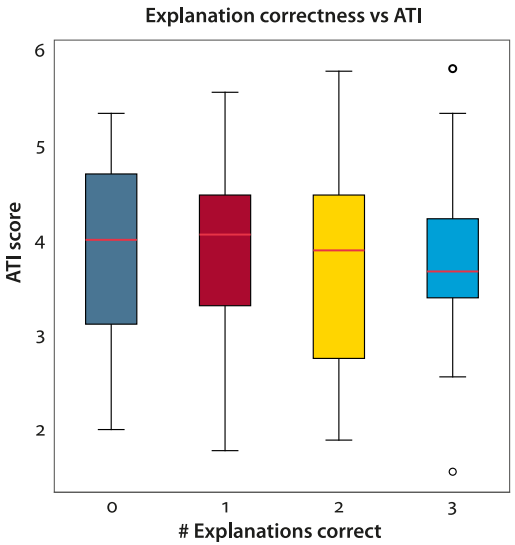

Figure 15. Boxplot of the ATI scores grouped by number of correct explanations

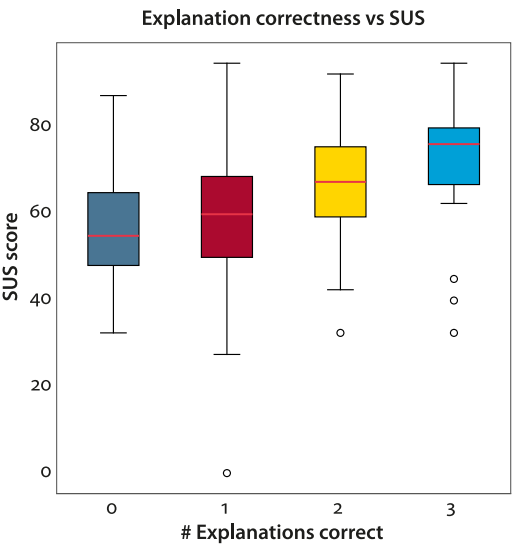

Figure 16. Boxplot of the SUS scores grouped by number of correct explanations

Table 9. Contingency tests for the godspeed and SUS scores, grouped by number of correctly explained errors. Entries are: F-/H-statistics and p-values of One-Way ANOVA respectively Kruskal-Wallis. p-values < 0.05 are highlighted in **bold**, p-values < 0.1 are highlighted in *italic*

ATI		
Shapiro-Wilk (p-value = 0.1401)		
One-Way ANOVA (F-statistic = 0.7618, p-value = 0.5177)		
Anthropomorphism		
Shapiro-Wilk (p-value = **2.1451e-05**)		
Kruskal-Wallis (H-statistic = 6.2771, p-value = 0.0989)		
Condition 1	**Condition 2**	**p-value**
0	1	0.3271
	2	0.4992
	3	0.0253
1	2	0.7268
	3	*0.0640*
2	3	0.047
Likeability		
Shapiro-Wilk (p-value = **0.0125**)		
Kruskal-Wallis (H-statistic = 1.1651, p-value = 0.7614)		
Perceived Intelligence		
Shapiro-Wilk (p-value = 0.2516)		
One-Way ANOVA (F-statistic = 0.5137, p-value = 0.6736)		
SUS		
Shapiro-Wilk (p-value = **0.0168**)		
Kruskal-Wallis (H-statistic = 13.9733, p-value = **0.0029**)		
Condition 1	**Condition 2**	**p-value**
0	1	0.4658
	2	0.0669
	3	0.0024
1	2	0.1019
	3	0.0012
2	3	0.1094

6. Discussion

Based on the online survey results (cf. Section 5), we can derive statements regarding our hypotheses. The **first hypothesis** stated that providing insights on the robot's architecture will improve knowledge and understanding of such. The results from the feature listing show that architecture information increases the

knowledge about the robot's hardware. In particular, this is true for features that might not be well-known, like the *microphone* or the *pan-tilt unit*. Already familiar features are not boosted. Apart from that, the architecture instructions did no improve the understanding of the software features. Instead, we observed the opposite: the architecture condition mentioned fewer software features than the other conditions. This problem might have occurred based on the design of the instruction video. The video mentioned each hardware feature before the software side usage. Therefore, an overload of information could have led to only memorizing the hardware features. Besides, the video may have primed users, leading to an increased focus on information from the instruction video.

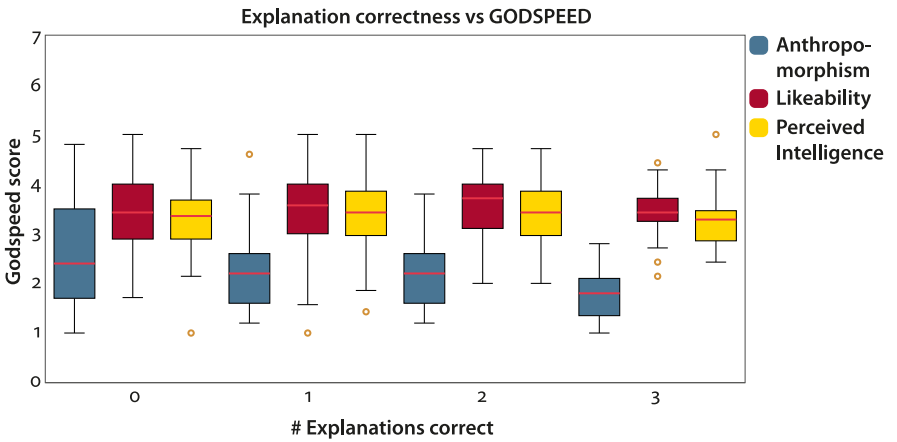

Figure 17. Boxplot of the GODSPEED scores grouped by number of correct explanations

Our **second hypothesis** stated that architecture insights help to recognize and explain errors in human-robot interactions, as these improve the users' mental model about the robot. We tested this hypothesis with two different approaches to communicate architecture information. Our results were not able to support our hypothesis when only an instruction video presents information. The provided architecture information does not improve the ability to explain errors in human-robot interaction. A reason might be a lack of insights reading the current status of the robot. Although users have an improved initial mental model, they cannot apply the architecture knowledge in the current situation. Similarly, the visualization approach did not improve the ability to explain errors. Instead, it seems to make it worse for some error types (i.e., `Failed Speech` and `Failed SM`). An overload of information and misinterpretations might be the problem. In turn, the quality and intuitiveness of our visualization might have produced this. Additionally, a visualization without further information on what it communicates allows

much freedom for interpretation. Consequently, this can lead to confusion, which results in unexplainable error situations.

However, both approaches seem to complement each other. The results show that in the fourth condition of showing an instruction video and providing a visualization, the ability to detect errors is significantly higher. By improving the mental model through the instruction video, the visualization is less confusing while its meaning can be more easily recognized. Additionally, the visualization helps to observe the learned architecture information throughout the human-robot interaction. Which in turn leads to an improved ability to detect and explain erroneous interactions.

Additional support for our hypothesis regarding the more appropriate shaped mental model (i.e., not equal to the mental model for another human) provides the comparison of the questionnaire scores for the number of correctly explained errors. The *anthropomorphism* score is lower for participants who could explain all three errors. Furthermore, the *SUS* score is higher for participants with more correctly described errors. It suggests that these participants ascribed less human-like characteristics to the robot while rating the system as useful despite its apparent limitations. Furthermore, we compared the technical affinity of the participants who could or could not explain the errors. The results did not show any significant differences. Therefore, our strategies of communicating insights of the robot seem to be understandable for people independent of their technical affinity, including naive users.

While we can see that architecture information improves the ability to explain errors in interaction and therefore improves the user's mental model, this ability also depends on the concept where the error occurs. The results from our **third hypothesis** showed that participants explained the object recognition error more frequently than the speech recognition or the state machine. The better observability and familiarity of the object recognition concept explains this. More than half of the participants stated that they had observed the object recognition error in the interaction itself. The speech recognition and state machine errors, on the other hand, were detected by reference to the video of the previous correctly working interaction. We assume that despite theoretical knowledge about technical concepts, users need to observe the explicit structure of the interaction to identify an error.

These results suggest that it is reasonable to communicate technical concepts to users, to improve their mental model. Furthermore, we showed that the used concepts have a significant influence on the error detection ability. For example, we selected the concept of object recognition in a way that people could use previous knowledge of QR-scanners. Therefore, we argue that already established technical concepts should gear the design of robotic concepts, together with tech-

niques to communicate it. Although we showed that communicating technical concepts helps to shape a more appropriate mental model, we cannot make a general statement about the design of such information. Especially the cognitive load of users should be considered. Further studies are required to evaluate how much information about a concept contributes to a better understanding of them. Moreover, each concept has specific factors that a user needs to understand.

Our observations show that the concept of finite state machines, which is at the core of human-robot interactions, is neither familiar nor observable. Hence, further improvements towards this direction are needed. It is reasonable to consider the observations regarding pragmatic frames (cf. Section 2.2). Vollmer et al. (2016) investigated the use of pragmatic frames in human-robot interactions. They showed that an interaction with a robot, where the user teaches the robot, is still not flexible enough for intuitive interactions with the system. We suggest that an alternative framework should allow for simple reasoning on the robot's side to communicate its internal processes. Simultaneously, such a framework needs to be more flexible and adaptable for naive users. Thus, enabling a more intuitive interaction. In that way, the mental model develops while shaping the robot's behavior, based on the user's expectations. Kaptein et al. (2017) used a *belief-desire-intention (BDI)* based agent to communicate the decision making process. While they showed that adults prefer explanations in terms of the robot's goals, we believe that this forms an incorrect mental model about the robot with unrealistic expectations regarding its cognitive abilities. Besides, a *BDI* architecture with its hierarchical structure might not be suitable to be taught by naive users, taking into account the potential complexity of such structures. Saunders et al. (2015) developed a flexible system that users can personalize. Current sensor states guide the decision-making process, similar to reactive behaviors. Such a system is based on technical concepts but probably still simple enough for humans to understand. Thus, it could present a more intuitively understandable alternative to currently widely used state machine frameworks.

7. Conclusion

This paper is concerned with the problem, how to improve users' mental models of robots in human-robot interactions. We investigated a new approach by communicating insights into the robot's architecture in two different ways. One method tried to communicate technical concepts in a direct way, similar to a manual. The second approach was indirect by visualizing the current internal states of the robot.

We evaluated the approaches in an online survey, where participants had to detect and further elaborate erroneous human-robot interactions. In addition to each method, we compared them to a baseline and also as a combined approach.

The results showed that both approaches on their own do not help to explain erroneous situations. In contrast to this, the combination of both strategies improved the ability to detect errors significantly. Furthermore, the ascribed anthropomorphic characteristics were lower, while the usability was rated higher.

Based on the results, we conclude that the current social robotics trend, pretending that the robot has human-like abilities and emotions, might not be the optimal research direction. Our society is confronted with technology long enough for people to be able to deal with technical concepts. The results of this paper suggest that robot designers should communicate the technical concepts to the users. Thereby, robots and users can achieve common ground on mental models.

8. Future work

For our further work, we will develop and evaluate new technical concepts for robotics. Thus, we want to improve users' mental models of the robot. In turn, this will potentially optimize human-robot interactions and reduce the occurrence of erroneous interactions.

One finding was that the state machine concept was one of the most incomprehensible ones for participants. Because the robot control is at the center of each interaction, users should understand this concept to detect and correct errors in interactions. Therefore, we will focus on alternatives to this concept.

Funding

This work was funded by the Honda Research Institute Europe, 63073 Offenbach am Main, Germany.

References

Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An empirical evaluation of the system usability scale. *Intl. Journal of Human-Computer Interaction*, *24* (6), 574–594. https://doi.org/10.1080/10447310802205776

Bartneck, C., Kulié, D., Croft, E., & Zoghbi, S. (2009). Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *International journal of social robotics*, *1* (1), 71–81. https://doi.org/10.1007/s12369-008-0001-3

Beller, W. E., & Wang, Y. P. (1997). Bar code dataform scanning and labeling apparatus and method [US Patent 5,602,377].

Brand, R. J., Baldwin, D. A., & Ashburn, L. A. (2002). Evidence for 'motionese': Modifications in mothers' infant-directed action. *Developmental Science*, *5*(1), 72–83. https://doi.org/10.1111/1467-7687.00211

Breazeal, C., Dautenhahn, K., & Kanda, T. (2016). Social robotics. *Springer handbook of robotics* (pp. 1935–1972). Springer. https://doi.org/10.1007/978-3-319-32552-1_72

Breazeal, C., Kidd, C., Thomaz, A., Hoffman, G., & Berlin, M. (2005). Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 708–713. https://doi.org/10.1109/IROS.2005.1545011

Breslow, N. (1970). A generalized kruskal-wallis test for comparing k samples subject to unequal patterns of censorship. *Biometrika*, *57* (3), 579–594. https://doi.org/10.1093/biomet/57.3.579

Bruner, J. (1985). Child's talk: Learning to use language. *Child Language Teaching and Therapy*, *1* (1), 111–114. https://doi.org/10.1177/026565908500100113

Cakmak, M., & Takayama, L. (2014). Teaching people how to teach robots: The effect of instructional materials and dialog design. *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, 431–438. https://doi.org/10.1145/2559636.2559675

Clement, J. (2020). Most popular mobile messaging apps worldwide as of october 2019, based on number of monthly active users [Retrieved: 2020-06-09, from https://www.statista.com/statistics/258749/most-popular-global-mobile-messenger-apps/\#statisticContainer].

de Greeff, J., & Belpaeme, T. (2015). Why robots should be social: Enhancing machine learning through social human-robot interaction. *PLOS ONE*, *10* (9), 1–26. https://doi.org/10.1371/journal.pone.0138061

Duffy, B. R. (2006). Fundamental issues in social robotics. *International Review of Information Ethics*, *6* (12), 2006. https://doi.org/10.29173/irie137

Dunn, O. J. (1964). Multiple comparisons using rank sums. *Technometrics*, *6* (3), 241–252. https://doi.org/10.1080/00401706.1964.10490181

Franke, T., Attig, C., & Wessel, D. (2019a). A personal resource for technology interaction: Development and validation of the affinity for technology interaction (ati) scale. *International Journal of Human-Computer Interaction*, *35*(6), 456–467. https://doi.org/10.1080/10447318.2018.1456150

Franke, T., Attig, C., & Wessel, D. (2019b). A personal resource for technology interaction: Development and validation of the affinity for technology interaction (ati) scale. *International Journal of Human-Computer Interaction*, *35* (6), 456–467. https://doi.org/10.1080/10447318.2018.1456150

Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F. J., & Marién-Jiménez, M. J. (2014). Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, *47*(6), 2280–2292. https://doi.org/10.1016/j.patcog.2014.01.005

Hamacher, A., Bianchi-Berthouze, N., Pipe, A.G., & Eder, K. (2016). Believing in bert: Using expressive communication to enhance trust and counteract operational error in physical human-robot interaction. *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 493–500. https://doi.org/10.1109/ROMAN.2016.7745163

Hassenzahl, M., Borchers, J., Boll, S., Pütten, A.R.-V.D., & Wulf, V. (2020). Otherware: How to best interact with autonomous systems. *Interactions*, *28*(1), 54–57. https://doi.org/10.1145/3436942

Hegel, F., Gieselmann, S., Peters, A., Holthaus, P., & Wrede, B. (2011). Towards a typology of meaningful signals and cues in social robotics. *2011 RO-MAN*, 72–78. https://doi.org/10.1109/ROMAN.2011.6005246

Hindemith, L., Vollmer, A.-L., Wrede, B., & Joublin, F. (2019). Pragmatic frames as an approach to reduce misinterpretations in human-robot-interaction. *Proc. Int. Conf. on Development and Learning (ICDL-EPIROB)*.

Kaptein, F., Broekens, J., Hindriks, K., & Neerincx, M. (2017). Personalised self-explanation by robots: The role of goals versus beliefs in robot-action explanation for children and adults. *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 676–682. https://doi.org/10.1109/ROMAN.2017.8172376

Kwon, M., Huang, S.H., & Dragan, A.D. (2018). Expressing robot incapability. *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 87–95. https://doi.org/10.1145/3171221.3171276

Liu, S. (2020). Global market share held by operating systems for desktop pcs, from january 2013 to january 2020 [Retrieved: 2020-06-09, from https://www.statista.com/statistics/218089/global-market-share-of-windows-7/].

McCracken, D.D., & Reilly, E.D. (2003). Backus-naur form (bnf). *Encyclopedia of computer science* (pp. 129–131). John Wiley; Sons Ltd.

Nelson, D.G.K., Hirsh-Pasek, K., Jusczyk, P.W., & Cassidy, K.W. (1989). How the prosodic cues in motherese might assist language learning. *Journal of child Language*, *16* (1), 55–68. https://doi.org/10.1017/S030500090001343X

Otero, N., Alissandrakis, A., Dautenhahn, K., Nehaniv, C., Syrdal, D.S., & Koay, K.L. (2008). Human to robot demonstrations of routine home tasks: Exploring the role of the robot's feedback. *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 177–184. https://doi.org/10.1145/1349822.1349846

Pitsch, K., Vollmer, A.-L., Rohlfing, K.J., Fritsch, J., & Wrede, B. (2014). Tutoring in adult-child interaction: On the loop of the tutor's action modification and the recipient's gaze. *Interaction Studies*, *15*(1), 55–98. https://doi.org/10.1075/is.15.1.03pit

Rahwan, I., Cebrian, M., Obradovich, N., Bongard, J., Bonnefon, J.-F., Breazeal, C., Crandall, J.W., Christakis, N.A., Couzin, I.D., Jackson, M.O., et al. (2019). Machine behaviour. *Nature*, *568*(7753), 477–486. https://doi.org/10.1038/s41586-019-1138-y

Rohlfing, K.J., Wrede, B., Vollmer, A.-L., & Oudeyer, P.-Y. (2016). An alternative to mapping a word onto a concept in language acquisition: Pragmatic frames. *Frontiers in psychology*, *7*, 470. https://doi.org/10.3389/fpsyg.2016.00470

Saunders, J., Syrdal, D.S., Koay, K.L., Burke, N., & Dautenhahn, K. (2015). "teach me-show me" – end-user personalization of a smart home and companion robot. *IEEE Transactions on Human-Machine Systems*, *46* (1), 27–40. https://doi.org/10.1109/THMS.2015.2445105

Schillinger, P., Kohlbrecher, S., & von Stryk, O. (2016). Human-robot collaborative high-level control with application to rescue robotics. *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2796–2802. https://doi.org/10.1109/ICRA.2016.7487442

Schulte, C., & Budde, L. (2018). A framework for computing education: Hybrid interaction system: The need for a bigger picture in computing education. *Proceedings of the 18th Koli Calling International Conference on Computing Education Research*, 1–10. https://doi.org/10.1145/3279720.3279733

Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, *52*(3–4), 591–611. https://doi.org/10.1093/biomet/52.3-4.591

Soon, T. J. (2008). Qr code. *Synthesis Journal*, *2008*, 59–78.

Staggers, N., & Norcio, A. F. (1993). Mental models: Concepts for human-computer interaction research. *International Journal of Man-machine studies*, *38*(4), 587–605. https://doi.org/10.1006/imms.1993.1028

Stanford Artificial Intelligence Laboratory et al. (2014, July 22). *Robotic operating system* (Version ROS Indigo Igloo). https://www.ros.org

Sterelny, K. (1990). *The representational theory of mind: An introduction*. Basil Blackwell.

Sweller, J., van Merriënboer, J. J., & Paas, F. (2019). Cognitive architecture and instructional design: 20 years later. *Educational Psychology Review*, 1–32. https://doi.org/10.1007/s10648-019-09465-5

Thomaz, A. L., & Cakmak, M. (2009). Learning about objects with human teachers. *2009 4th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 15–22. https://doi.org/10.1145/1514095.1514101

Vollmer, A.-L., Lohan, K. S., Fritsch, J., Wrede, B., & Rohlfing, K. (2009). Which motionese parameters change with children's age?

Vollmer, A.-L., Mühlig, M., Steil, J. J., Pitsch, K., Fritsch, J., Rohlfing, K. J., & Wrede, B. (2014). Robots show us how to teach them: Feedback from robots shapes tutoring behavior during action learning. *PloS one*, *9* (3). https://doi.org/10.1371/journal.pone.0091349

Vollmer, A.-L., & Schillingmann, L. (2018). On studying human teaching behavior with robots: A review. *Review of Philosophy and Psychology*, *9* (4), 863–903. https://doi.org/10.1007/s13164-017-0353-4

Vollmer, A.-L., Wrede, B., Rohlfing, K. J., & Oudeyer, P.-Y. (2016). Pragmatic frames for teaching and learning in human-robot interaction: Review and challenges. *Frontiers in neurorobotics*, *10*, 10. https://doi.org/10.3389/fnbot.2016.00010

Wortham, R., Theodorou, A., & Bryson, J. (2017). Robot transparency: Improving understanding of intelligent behaviour for designers and users, 274–289. https://doi.org/10.1007/978-3-319-64107-2_22

Address for correspondence

Lukas Hindemith
CITEC 1.224
Universität Bielefeld
Inspiration 1
33619 Bielefeld
Germany
lhindemith@techfak.uni-bielefeld.de

Biographical notes

Lukas Hindemith studied intelligent systems at Bielefeld University. He received the Master's degree from Bielefeld University, Germany, in 2019. He then joined the Applied Informatics Group and the Research Institute for Cognition and Robotics (CoR-Lab) at Bielefeld University, Germany, to work in a collaborated project together with the Honda Research Institute Europe GmbH. His work focuses on mismatches in mental models in human-robot interaction.

Jan Philip Göpfert is a final year PhD candidate in the Machine Learning Group at Bielefeld University, under the supervision of Prof. Dr. Barbara Hammer. He researches uncertainty in Machine Learning, generative models, and adversarial robustness.
jgoepfert@techfak.uni-bielefeld.de

Christiane B. Wiebel-Herboth studied psychology at the University of Tüabingen before she received her phD on material perception in the lab of Karl Gegenfurtner at the University of Giessen. After a post-doc on lightness perception with Marianne Maertens at the TU Berlin she started to work as a senior scientist at the Honda Research Institute Europe in Offenbach, where she works now in the field of human-machine interaction research.
christiane.wiebel@honda-ri.de

Britta Wrede is head of the Medical Assistive Systems Group at the Medical School OWL at Bielefeld University. After receiving her M.A. and PhD title from the Faculty of Linguistics in 1999 and the Technical Faculty in 2002 respectively, she spent one year as DAAD fellow at the International Computer Science Institute (ICSI) in Berkeley. After rejoining Bielefeld University working on different projects, she became head of the Applied Informatics Group in 2009. Her research on developing assistive systems for therapy and diagnostics support is based on the hypothesis that assistance needs to be embedded in social interaction.
bwrede@techfak.uni-bielefeld.de

Anna-Lisa Vollmer is currently a postdoctoral researcher at Bielefeld University, Germany. After receiving a PhD from Bielefeld University, she was an Experienced Researcher in the MSCA ITN RobotDoC at Plymouth University, UK and was awarded a Starting Research Position at INRIA Bor-deaux/ENSTA ParisTech, France. Her research interests include robot learning in social interaction, human-in-the-loop machine learning, and co-constructed communication.
avollmer@techfak.uni-bielefeld.de

Publication history

Date received: 19 June 2020
Date accepted: 25 November 2021