Natural speaking (inter)faces for cars

Valentin Nicolescu[1], Jan Marco Leimeister[2], Helmut Krcmar[1]

[1]Lehrstuhl für Wirtschaftsinformatik, Technische Universität München
[2]Fachgebiet Wirtschaftsinformatik, Universität Kassel

1 Introduction

Modern upper class vehicles provide an increasing number of functions, the purpose of which is to make travelling by car more pleasant and safer for passengers. On the one hand, the quantity of functions require car manufacturers to integrate the corresponding control elements; on the other hand, the driver is expected to become familiar with the use of the new or sometimes even hidden secondary functions of these control elements. These hidden functions are due to the limited space on the dashboard, making the combination of different functions in one control element necessary. Pooling functions together in a control element increases the risk that drivers will not be aware of the features of a vehicle. Thus, DaimlerChrysler removed about 600 such hidden functions with a model change, since they were not known and hence were not being used by drivers (Koch 2004). If the future number of functions in a car continues to grow, the possibilities of combined control elements and hidden functions will have been exhausted.

Another challenge that car manufacturers and their customers have to face is the increasing number of software based functions, particularly in secondary and tertiary task areas, i.e. the use of comfort or entertainment devices such as air conditioning, media centers or navigation systems (Tönnis et al. 2006). The application of software development concepts for automotive services requires drivers to have full control of the new functions. These new automotive services also need more information about the driver in order to tap full potential. For example, an electric car will need information about the driving behavior of a user to develop its battery loading strategy.

This makes a new approach necessary for the in-vehicle user interface that enables easy access to a large number of existing and/or future functionality, as well as information without distracting the driver from driving.

The optimal solution for this issue would be the permanent presence of a co-driver-in-the-box who could recite all user manuals of the car. This co-driver would also adapt to the driver's behavior and provide information in crucial situations whenever the driver requested assistance. The information would be suitable to the situation and be provided in the right amount at the right point in time.

This article provides a general understanding of this vision by elaborating on the experiences of working with a prototype in a series-production vehicle.

2 Analyzing existing technologies

Regarding technologies available today for this futuristic vision, several advancements can be identified towards the realization of such a system.

For many years, avatars have been used in web and offline applications to incorporate virtual dialog partners. In most cases they have the task of virtual assistants guiding users to adequate information or products from of a variety of goods. Thus, an Embodied Conversational Agent (ECA) (Cassell 2000) uses social interaction to build up a trusting relationship with the user. This is possible due to the user's perception of the avatar as a virtual human impersonating the corresponding application or system (Badler 1997).

Such ECAs communicate with the user on the Internet usually via text input and output. Offline ECAs also use separate input devices or speech recognition, if applicable. The logical component for finding and generating the right answers is often realized in a so-called chatbot. Chatbots are software applications that are based on the idea of the Turing test, which simulates a real interlocutor (Turing 1950).

This can be achieved by the use of artificial intelligence in the strict sense or through the implementation of case-based reasoning, considered to be artificial intelligence in a wider sense. Therefore, a previously created database is searched for a suitable output to a user input, whereas situational factors can influence the response generation.

Apart from the application of such logical components in conjunction with avatars, chatbots are also used in speech dialog systems. In doing so, there is no representation by an avatar, but language as a new modality is added.

For these voice systems, such as telephone information systems, the user's speech is analyzed, converted into text and the corresponding response is then rendered using pre-recorded or synthesized speech. While online systems usually do not use speech because of the limited bandwidth and computing power necessary for real-time speech synthesis, numerous offline ECAs can be observed to have an acoustic output (Ball et al. 1997; Cassell et al. 2001). The response of the chatbot is put in the avatar's mouth, which ideally has lip sync. However, speech recognition in conjunction with an avatar can be found only at a few experimental systems (Cassell 2001).

To display the avatars, as well as the visualization of information which are presented in response to a question about the car, a display component is necessary (Paivio 1979). This has to be able to display not only 2D graphics and animations, but also 3D objects consisting of the avatar itself and its virtual environment. The technology for real-time rendering of 2D and 3D objects has been developed further, specifically influenced by the computer games industry. The challenges of using such a 3D engine in the car are the limited computing power and display size that are available in current vehicles.

To fully consider the technological aspects of the vision presented, one needs to consider how the behavior of the driver can be identified and interpreted. Basically, there are two different approaches: the observation of the driver himself and the observation of the effects that the behavior of the driver has. The driver can be directly surveyed by gesture, speech or voice recognition, as well as pulse and temperature sensors. Apart from surveying the driver directly, the impact of his actions and his environment can be analyzed as well. Most of these technologies are still in an experimental stage regarding the use in vehicles and currently cannot be found in mass-production. Modern medium and upper class vehicles have available various bus systems and numerous sensors that perceive the environment of the vehicle. The bus systems combine several vehicle control devices inside the vehicle and allow all to share the same information. For example, a bus system can provide information about the states of all the control elements and instruments, while another bus distributes information on the drive train and the chassis frame. The partitioning of different buses varies, depending on the automobile manufacturer, but most use a Controller Area Network (CAN). With this information, it is possible to see if the user operates a control element, if another vehicle comes too close, if a technical defect occurs, and much more.

In addition to these core technologies of an ECA, many other options can be identified when designing such a system independently from the future application area. To obtain reference architecture with all possible design options of an ECA, we analyzed and disassembled 34 different ECA and speech dialog systems in their individual components. After this, we consolidated these components in order to transpose them to a new framework for the architecture of ECAs (see Figure 1.)

Thereby, four functional areas have been identified: input, output, response determination and design (elaborated upon in Section 3). The inputs for the ECA in the form of voice, text or other options are physically collected. Next, the digital signals are converted and translated in a first semantic transformation, for example from acoustic signals into text. These transformed inputs are processed in the input manager, netted and transmitted in a consolidated response to the reaction determination.

Depending on whether a chatbot or an agent is used for the reaction determination, different components can be applied for reaction determination (McTear 2002). In both cases, the response determination will be coordinated by the dialogue manager, determining how a particular input has to be processed. The determined reaction is synchronized by the presentation manager and then forwarded to the corresponding output channels.

The knowledge base or the goal system of the reaction determination, as well as the avatar itself and the modeling of its environment, are maintained independent of the runtime environment in the design phase.

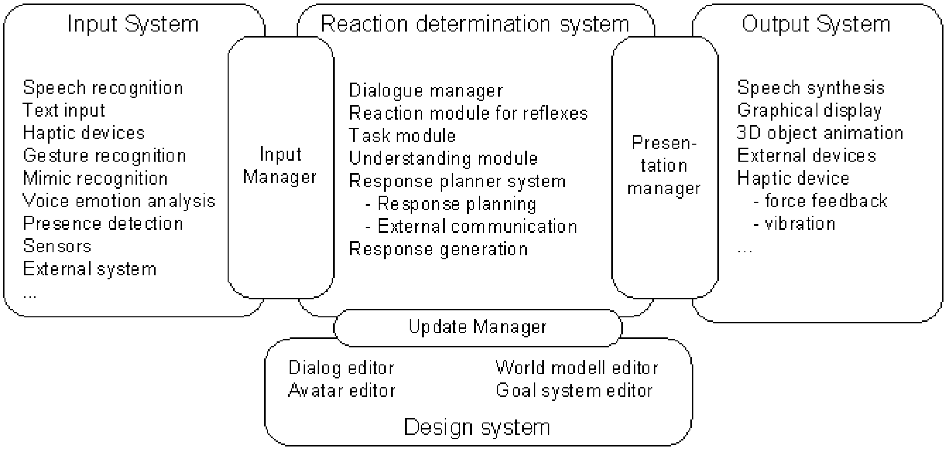

Figure 1: Reference architecture for embodied conversational agents in general

3 Recombining technology for a new domain

The identified technical components can be applied to an ECA in a vehicle only in a limited way, since many technologies are neither present nor suitable for the use in a vehicle. In addition, it is advisable to start with basic functions and to enhance the first prototype later with new options. Our prototype, which realizes the vision presented above, was implemented in a mass-production car of type Audi A4 model 2008. Since the central unit of the vehicle, called head unit, consists of a proprietary hardware and software platform, an additional computer in the form of a standard PC with the mini ITX form factor was built in for experimental purposes.

Apart from the characteristics of a conventional PC, the computer built-in model features a standard CAN-bus card and a solid state disk. This ensures the connection to the vehicle bus and also avoids issues caused by shocks during driving. A signal converter was used to connect the additional PC's video output to the series-integrated display with a resolution of 480x200 pixels. Thus, the video signal of the additional computer can be routed to the built-in display. In addition, the built-in microphone and speakers of the vehicle are attached to the computer. Based on this additional platform, it is possible to run standard software which could contain the presented core technologies. For our prototype, we implemented the software core components in OSGi, a Java framework, and tried to integrate standard software where available.

The input system of the prototype is characterized by speech input, which usually represents the only human modality in cars that is not already used to control the vehicle. Speech recognition in our prototype was activated by pushing the so-called push-to-talk-button in the steering wheel, thus avoiding the issue of voice activity detection. The applied speech recognition uses a predefined language grammar, which contains all discernible phrase patterns in a flexible way. Our prototype used a language grammar containing over 3000 different phrases patterns to identify spoken words. The activation of the push-to-talk-button is detected by the connection to the CAN-bus of the vehicle. This connection also allows the response to the operation of other controls, or certain events of the vehicle, such as a specific speed threshold or an error message.

The output area is particularly characterized by the real-time computed 3D avatar, which is embedded in a virtual 3D environment. This component is controlled by commands which are included in the responses and which are similar to a screenplay; for example, at one specific point in the text the avatar or any other 3D object will be animated in a certain way. In addition to the representation of 3D objects, 2D images and image sequences can be displayed, too. Tthe animations of the avatar, as well as 2D and 3D media, are stored in libraries that can be maintained in the design phase. Texts, as part of the answer, are forwarded to the avatar component, which speak the text lip-sync to the animation of the avatar.

The avatar component is embedded in a GUI framework that has the task to display status information and icons. The GUI framework is adopted to resemble a web browser in its look and feel. Further, the GUI framework takes over the task of the presentation manager and coordinates the transmission of data to the user interface and the avatar component. Figure 2 shows a screenshot of the GUI framework with the avatar component, the virtual environment and corresponding symbols as they can be seen in the vehicle.

The control of vehicle functions in the secondary and tertiary area over the CAN bus has deliberately not been implemented for this prototype, although it is technically possible by the CAN-bus connection. On the one hand, this decision is based on the experimental state of the system which, despite extensive tests, still has errors. Thus, these errors could have a negative impact on controllable functions and cause safety-critical behavior. On the other hand, the later insertion of a control device in an already adjusted CAN bus architecture could cause the failure of the entire bus, also known as the "babbling idiot problem" (Anderl 2005).

The reaction determination embeds a chatbot which uses a knowledge base with information about the response generation regarding all available output channels. The answers given by the chatbot resemble screenplays which contain text to be synthesized, animations, media and control information for the GUI framework. The maintenance of this extensive knowledge base happens in the design phase using the dialogue designer (described in detail later).

The transfer of inputs to the chatbot is done by the dialog manager, who decides whether an input will be processed regularly via the chatbot or processed with high priority. A prioritized processing represents reflexes of the system, for example when the driver presses the push-to-talk-button. In this case, immediately after pressing the button, a corresponding icon in GUI framework appears signaling that the ECA is processing the voice input. Alternatively, an avatar animation can be called, indicating that the avatar is listening.

The design system of the prototype includes a dialogue designer, which allows an integrated maintenance of the knowledge base with all available output options. A conventional commercial avatar and a 3D object editor with a special plug-in allows adjustment of the avatar itself, its animations, and 3D objects. The maintenance of 2D media is also accomplished by commercially available software. Given the use of a chatbots for the reaction determination, the world model and goal system editors are not necessary.

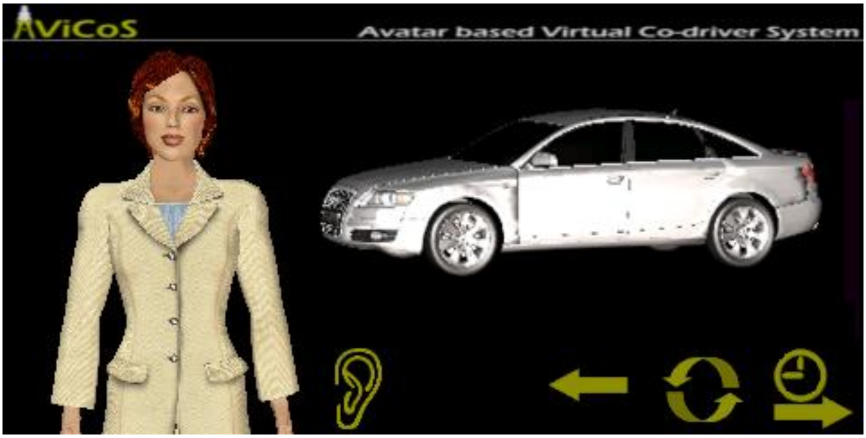

Figure 2: Screenshot of the prototype's frontend

4 Use cases of the ECA

The use of the prototype in the vehicle can be done in two different ways: by first asking for specific information and or by explorative use of control elements in the secondary and tertiary sector. The first use case is designed as a substitute for the user's manual and allows access to information about features and functions of the car, also while driving. In addition to the information of the manual, the responses of the ECA are supported by media that could not be integrated in a print medium. This can happen by displaying a figure or a 3D object on any topic, or the operation of control elements can be presented by the avatar using animations of the corresponding control elements.

A typical example for this use case is to ask about the operation of an assistance system in the vehicle. In the test carrier that we used, many manufacturer specific assistance systems, such as Audi lane assist or Audi drive select, had been integrated. At the driver's question, "How does Audi drive select work?" the avatar gives a general introduction to this feature, followed by an explanation of the importance and handling of its control elements. Since the explanation contains a large quantity of new information for a driver, it is subdivided into separate information chunks (Miller 1956). Each chunk represents a brief and compressed part of the explanation regarding a specific aspect.

Thematically related chunks are presented sequentially, whereby the driver can control how long the pause between two chunks is. He also has the possibility to go to the next or previous chunk manually. Similar to a web browser, he also has the option of repeating the current chunk, so he can hear and see it again if he did not understand it the first time. Additional icons in the GUI framework indicate to the driver if more detailed information is available on a chunk, or if it is part of a tutorial. For tutorials, the user can directly apply what the avatar has explained by using the appropriate control element. The ECA gives the driver feedback on the use of a control element to confirm or to correct his actions.

However, the driver does not have to follow the proposed sequence of information and can stop the presentation at any time, or he can issue a further question and thus jump to a new topic. After the chunks of a specific topic have all been played, whether automatically or manually, the user is offered to be redirected to other related topics. These described handling paradigms are valid for both use cases.

The specific questions about a function and its properties require knowledge about the existence of a certain available function. In unfamiliar vehicles it often happens that the driver is unfamiliar with the vehicle's features, as can happen when renting a car and the prominent or important functions are not user-friendly.

For such cases, the so-called 'Touch & Tell' mode has been implemented, in which any element of the secondary or tertiary functional area can be operated by the driver after activating this mode. By the operation of a control element, the ECA presents a brief explanation of the function used and offers the option of more detailed information. When activating the Touch & Tell mode, the avatar points out that the activation of a specific control element leads to an explanation, but also activates the associated function of the car. For security reasons, this restriction has to be tolerated in the current prototype, since there is no separation between triggering a function and its operation.

Thus, the Touch & Tell mode gives an overview of important features of the vehicle in a very short time and allows the drive to become acquainted very quickly with an unknown vehicle, albeit superficially. For longer use of the vehicle, specific questions on known functions are answered with detailed information, including hidden or secondary features of controls.

To ensure that the security of the driver is not compromised, two speed thresholds are integrated in both use cases. Starting at the first threshold of about 40 miles, instead of animations and 3D objects, a 2D still image is displayed to reduce the distraction. From the second threshold of about 80 miles onwards, the avatar disappears entirely, but the ECA can still be operated by speech.

5 Bringing the system to life

The implementation of the technical platform is only the first step towards a complete ECA. Although numerous technical options are available in the prototype, it can only be filled with life if the knowledge base is extensively maintained and is using these options. It is therefore the spark of life avatars and significantly affects how human the system is perceived to be by the user. Although the prototype does not cover the functionality for small talk and dialogues can run in a flexible way, there still is a need for a rough dialogue structure as well as differentiated inputs and outputs. An issue results from the lack of a structured approach for the conversion of structured content in the user manual into dialogues.

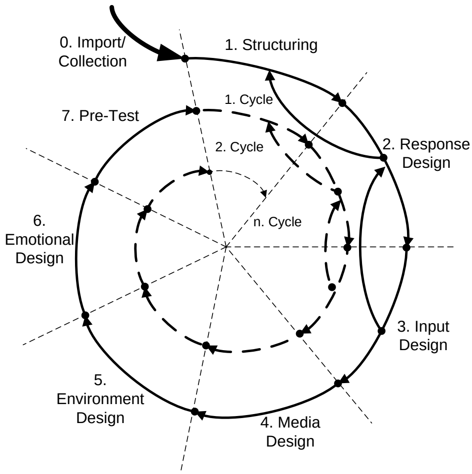

Figure 3: Process model for conversion of content into dialogues

This resulted in the need for a process model optimally supported by a software application that has been developed in parallel during the design of the knowledge base. The phase model, which is shown in Figure 3, consists of seven phases that will be processed sequentially in several cycles, if necessary. This is similar to the hermeneutic circle with each cycle approaching the complete coverage of all relevant content with the knowledge base. The fragmentation in phases and the clear definition of steps and interfaces between the phases allow the division of labor according to different roles. Thus, some phases can be processed by linguists or editors, while a different phase can be handled by graphic designers or psychologists. All phases of the model are supported by descriptions of specific steps and guidelines for the content development.

The initial import phase brings the structured contents into the software system. In the case of unstructured content instead of importing, it is collected directly in the system. The first cyclic phase represents structuring at which the automatically generated structure of the import is revised. Regarding the car's user manual, text sections are relevant for several car configuration variants. A different operation of the variants becomes clear only by reading the section. This undifferentiated structure of the manual must be clarified using variables in the dialogue for the corresponding configuration variants. For example, a section can be marked important for all types of automatic gearboxes. Only in the section of the operation of tiptronic and multitronic variants is it distinguished.

The response design is done in the second phase and includes the arrangement and linguistic revision. The texts, imported directly from the digital version of the user manual, contain many elements that will sound inappropriate in the speech synthesis. Thus, texts from brackets are resolved, abbreviation written out and direct addressing of the driver implemented. In this phase, the need for a structural adjustment may result, which is why kickbacks to the first phase are possible.

While the response design covers a finite amount of information, the quantity of possible user inputs can hardly be foreseen. Thus, great importance has to be attached to this phase, and it also takes the most time. Through detailed questions of the user at this point, kickbacks to the previous phase can also be necessary. Once the proper information, and thus the closely linked possible inputs are maintained, the responses can be enriched with different 2D and 3D media and adequate avatar animations in the fourth phase.

In the phase of the environment design, operation of basic technical functions through voice input is covered at the first cycle; for example, the user will be given the opportunity to adjust user options by voice. In the following cycles, in particular, user inputs which contain omissions have to be covered. Thus, if the ECA previously explained something regarding air conditioning and the driver asks, "Where is it" in this phase, it can be designed so that the ECA understands that in the current dialogue context, "it" refers to the air conditioning.

The emotional design covers the modeling of feelings on behalf of the avatar and the reactions to emotions of the driver. This phase is optional and makes sense only if the topic that the ECA is talking about is emotionally tainted or the expression of emotions supports the mediation of information. In the case of the developed prototype, no emotions have been integrated due to the technical facts that are taught.

The last phase is the test in which the adjustments of one cycle are tested; for example, approaches like the wizard-of-oz-test can be applied for these tests (Diaper and Shelton 1989).

6 Supporting the dialogue development with software

The creation and maintenance of the knowledge base is done with a component of the design system, the dialogue editor. It follows the paradigm of a model-driven architecture (Fettke and Loos 2003), after which a platform independent and thus flexible dialogue model is developed inside the editor (Cassell et al. 1998). In doing so, structured content can be imported; however, this platform independent model can also be exported in a platform dependent format. In the prototype, a digital version of the user manual is imported and the knowledge base with a matching language grammar is exported (Polifroni and Chung 2002).

The modeling of the knowledge base is carried out using a graphical dialog structure that can be edited by 'dragging & dropping' new elements into outputs or expected inputs, thus requiring no programming skills. Similarly, all technically possible output options such as synthesized speech, avatar animations or any media and 3D objects, can be integrated into an ECA's response. Inputs can include all possible formulations, whereas a synonym editor can automatically generate additional similar inputs on the base of inputs provided by the editor's user. Speech input and CAN bus messages can be maintained as possible inputs.
The generation of the response can be influenced dynamically by various logic elements at runtime. Variables can be used as a basis for branching within the dialogue structure. These variables can be set either by the driver's behavior or by CAN bus messages linked to a specific variable. Thus, situational responses can be generated at runtime.

7 User test of the prototype

To get some basic findings on the impact and acceptance of such an ECA in the vehicle, we interviewed and surveyed a total of 67 test persons as part of the evaluation. They were divided into three groups and each person had to deal with the same five tasks in the vehicle independent of the group. The participants of the first group used the user manual as a help, and thus served as a control group. The second group used the ECA as a tool in the vehicle standing still. The third group of participants used the ECA while they were driving. The test circuit of the third group consisted of approximately 60 miles on highways that could have been extended optionally to avoid pressure on the drivers. The subjects were pooled equally in the three groups, in accordance to the criteria of age and sex.

First, before performing the tasks, data was gathered in the form of a questionnaire. All participants were then accompanied by a moderator in the car who read the tasks to be performed in a step-by-step manner. The tasks dealt with the use of various functions in the secondary and tertiary function area, such as the activation of various assistance systems or the adjustment of air conditioning. After the completion of the five tasks, each participant received a second questionnaire which was different for group 1 and the other two groups, since the first group had no contact with the ECA. The processing of the tasks was recorded on video and was later analyzed for all three groups.

The analysis of the collected data was done in two ways: 1) a summative evaluation on all three groups, indicating how the ECA performed compared to the user

manual, and 2) a formative evaluation of the participants of groups two and three, where detailed examinations were carried out in order to identify possible areas to improve the ECA and its technical components.
When considering the benefits of the help systems, it became clear that the information that the ECA presented helped to identify and operate the specific functions in the vehicle (see Figure 4).

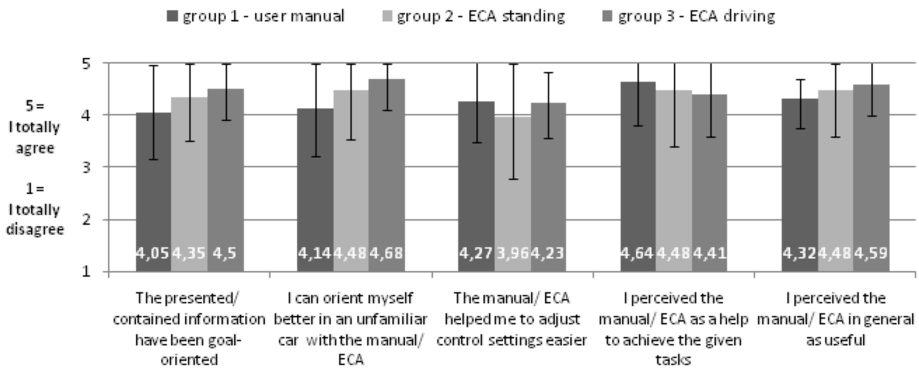

Figure 4: Evaluation of the manual's respectively the ECA's benefits (incl. standard deviation)

The participants perceived the ECA to be a better help in an unknown vehicle. An unclear picture was noticeable when asking about simplified operation. The questions regarding the usefulness of the ECA with respect to the user manual on the performance of the specific test tasks and the general helpfulness were answered controversially. The ECA was rated as being less helpful in respect to the concrete tasks, although it was rated as being very useful in general. Thus, it is clear that the operation of the prototype still needs improvement in handling issues; however, the principle of such a system was viewed as being helpful.

The overall assessment of the summative evaluation using school grades showed a similar result for the use of the user manual and the ECA in the driving car, while the ECA in the standing car was rated as being better.
As part of the formative evaluation of the ECA, the perceivable effects of technical components were evaluated in order to identify weaknesses. The components of the avatar, the content, the user interface, speech recognition, speech synthesis and dialogue history were considered. The user interface, the presented content and its visualization were rated particularly well. The worst rated component was the speech recognition, thus it will be a starting point for improvements. Although speech recognition rates of about 76% on average were achieved, this rate must continue to be improved.

8 Discussion

The reaction of users has shown that the idea of an ECA in a vehicle is perceived in a positive way even if the prototype is pointing out different areas for improvement. For example, in the video analysis we observed that users tried to use the push-to-talk-button in order to cancel the speech recognition. A functionality to pause during an explanation was missing as well.

In addition to the described handling issues, the question of the appropriateness of an avatar for such a system in a vehicle has to be faced. The avatar is not mandatory for basic information mediation. The avatar primarily has the task of a social actor (Reeves and Nass 1998), giving the driver the sensation that he can speak naturally to the car. Without a figurative illustration, it is expected that the user predominantly uses command words instead of natural language (Cheyer and Julia 1999). Using only command words, the driver holds back information that he would probably share when using natural speech. Due to the lack of information, the ECA can identify only with a low probability what the user wants and thus cannot respond with a reaction which the user may expect. This leads to increased frustration for the user, and thus the loss of acceptance for such systems. However, such ECA systems are necessary to handle a multitude of new functions in a meaningful way.

It is still open as to which shape ECAs will be implemented in mass-production cars one day. It can hardly be expected that the first production system of this type will already have such an extensive connection to the CAN bus or that 3D objects will be calculated in real-time. However, it can be expected that, compared to our prototype, even functionally limited implementations of such a system will provide added value.

One aspect of the prototype was not mentioned at all: the possible distraction of such a system and thus jeopardizing the safety of the driver. Depending on regional laws and similar to watching movies in the car, it can be expected that the animations of the avatar will be allowed in cars only while standing. Another aspect is the ability of the ECA not to be intrusive, i.e., speaking only when the user has requested it. To ensure this principle, a filter can be used to present a response immediately or later in a non-critical situation, depending on the calculated load of the driver.

9 Future research

On the basis of the prototype, numerous aspects in greater detail could be investigated. For example, it could be tested to find out which functions in the secondary and tertiary sector can be controlled by the ECA on the command of the user. It is possible that the driver's remark, “I'm cold” leads to an increased temperature of the air conditioning.

As part of the handling improvement in the area of speech recognition, voice activity detection could be investigated to replace the actuation of the push-to-talk-button (Singh and Boland 2007). It would also be possible to detect the mood in the speech in order to adjust the ECA's response: In the case of high tension, the response could be compressed, while for a relaxed voice, a more detailed explanation could be given.

The natural speech interaction of the dialogue could be used to collect information about the driver to build a user profile. For example, in the case of an electric car, the ECA could ask the driver, after turning off the car, whether he is planning to use the car again. Depending on the answer, the ECA could devise a strategy on how to load the battery the best way. Alternatively, when using the car regularly a user profiles could be automatically built. These profiles would also need to take into account that the car could be used by different drivers. For each driver an individual profile would need to be created, which determines, for example, also the appearance and behavior of the avatar. It is also expected that women will set up male avatars as virtual co-drivers (Nass et al. 1994).

While the presented prototype is dealing currently only with the contents of the user manual, it would also be possible to open up other use cases for such an ECA. The user manual is particularly interesting when a vehicle is new and unknown to the driver, but it is expected that this application may be used only rarely after a few months. In order to achieve constant use of the ECA, new use cases could be developed, such as assistants for environmentally friendly driving or an interactive manual on how to repair and maintain a car.

References

Anderl T (2005) Kommunikation für Fahrantriebe - Entwicklungsmethoden am Beispiel eines Hybridantriebs. Dissertation, Technische Universität München.

Badler N (1997) Real Time Virtual Humans, Proceedings of The Fifth Pacific Conference on Computer Graphics and Applications (pp. 4-14), Los Alamitos, CA: IEEE Computer Society Press.

Ball G, Ling D, Kurlander D, Miller J, Pugh D, Skelly T, Stankosky A, Thiel D, Van Dantzich M, Wax T (1997) Lifelike Computer Characters: The Persona Project at Microsoft. In: Bradshaw J (Ed.) Software Agents. AAAI/MIT Press, Menlo Park: 191-222.

Cassell J (2000) Nudge Nudge Wink Wink: Elements of Face-to-Face Conversation for Embodied Conversational Agents. In: Cassell J, Sullivan J, Prevost S, Churchill E (Ed.) Embodied Conversational Agents. The MIT Press, Cambridge: 1-28.

Cassell J (2001) Embodied Conversational Agents: Representation and Intelligence in User Interface. AI magazine, 22 (3): 67-83.

Cassell J, Bickmore T, Billinghurst M, Campbell L, Chang K, Vilhjálmsson H, Yan H (1998) An Architecture for Embodied Conversational Characters. In: Proceedings of the First Workshop on Embodied Conversational Characters, Tahoe City, California.

Cassell J, Vilhjálmsson HH, Bickmore T (2001) BEAT: the Behavior Expression Animation Toolkit. In: Proceedings of the Proceedings of SIGGRAPH 2001, Los Angeles, CA, USA.

Cheyer A, Julia L (1999) InfoWiz: An Animated Voice Interactive Information System. In: Proceedings of the Proceedings of the Workshop on Communicative Agents(Agents '99), Seattle.

Diaper D, Shelton T (1989) Dialogues with the Tin Man: A Natural Language Grammar for Expert System Naive Users. In: Peckham J (Ed.) Recent developments and applications of natural language processing. Kogan Page, London: 98-116.

Fettke P, Loos P (2003) Model Driven Architecture (MDA). Wirtschaftsinformatik, 45 (5): 555-559.

Koch B (2004) Zuverlässige Software fürs Auto. Fraunhofer Magazin, (4): 24-25.

McTear MF (2002) Spoken dialogue technology: enabling the conversational user interface. ACM Computing Surveys (CSUR), 34 (1): 90-169.

Miller GA (1956) The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. Psychological Review, 63 (2): 81-97.

Nass C, Steuer J, Tauber ER (1994) Computers are social actors. In: Proceedings of the Proceedings of the Conference on Human Factors in Computing Systems, Boston, Massachusetts, USA.

Paivio A (1979) Imagery and Verbal Processes. Lawrence Erlbaum Associates, Hillsdale, NJ.

Polifroni J, Chung G (2002) Promoting portability in dialogue management. In: Proceedings of the Proceedings of the ICSLP, Denver, Co.

Reeves B, Nass C (1998) The Media Equation - How People Treat computers, Television, and New media Like Real People and Places. CSLI Publications, Stanford, California.

Singh D, Boland F (2007) Voice Activity Detection ACM Crossroads, 13 (4):

Tönnis M, Broy V, Klinker G (2006) A Survey of Challenges Related to the Design of 3D User Interfaces for Car Drivers. In: Proceedings of the IEEE Virtual Reality Conference (VR 2006), Alexandria, Virginia, USA.

Turing A (1950): Computing machinery and intelligence. http://www.abelard.org/turpap/turpap.htm, 02.09.2006.