

# Chapter 2

## Development of Speech-Based In-Car HMI Concepts for Information Exchange Internet Apps

Hansjörg Hofmann, Anna Silberstein, Ute Ehrlich, André Berton, Christian Müller, and Angela Mahr

**Abstract** The permanent use of smartphones impacts the automotive environment. People tend to use their smartphone’s Internet capabilities manually while driving, which endangers the driver’s safety. Therefore, an intuitive in-car speech interface to the Internet is crucial in order to reduce driver distraction. Before developing an in-car speech dialog system to a new domain, you have to examine which speech-based human-machine interface concept is the most intuitive. This work in progress report describes the design of various human-machine interface concepts which include speech as main input and output modality. These concepts are based on two different dialog strategies: a command-based and a conversational speech dialog. Different graphical user interfaces, one including an avatar, have been designed in order to best support the speech dialog strategies and to raise the level of naturalness in the interaction. For each human-machine interface concept a prototype which allows for an online hotel booking has been developed. These prototypes will be evaluated in driving simulator experiments on usability and driving performance.

### 2.1 Motivation

The number of mobile Internet accesses has increased enormously within the last years. The permanent use of smartphones and their Internet capabilities also impacts the automotive environment. In order to be “always connected,” people tend to use their smartphone’s Internet access manually while driving. However, the manual

---

H. Hofmann (✉) • A. Silberstein • U. Ehrlich • A. Berton  
Daimler AG, Ulm, Germany  
e-mail: [hansjoerg.hofmann@daimler.com](mailto:hansjoerg.hofmann@daimler.com); [anna.silberstein@daimler.com](mailto:anna.silberstein@daimler.com);  
[ute.ehrlich@daimler.com](mailto:ute.ehrlich@daimler.com); [andre.bernton@daimler.com](mailto:andre.bernton@daimler.com)

C. Müller • A. Mahr  
DFKI, Saarbrücken, Germany  
e-mail: [christian.mueller@dfki.de](mailto:christian.mueller@dfki.de); [angela.mahr@dfki.de](mailto:angela.mahr@dfki.de)

use of smartphone distracts the driver from driving and endangers the driver's safety [4]. Therefore, the development of an intuitive and non-distractive in-car speech interface to the Web is essential in order to increase the driver safety [12].

Before developing a new speech dialog system (SDS) in a new domain, developers have to examine how users would interact with such a system. A previous Internet user study revealed that the human-machine speech interaction styles vary depending on the type of Internet activity [5]. The subjects were presented graphically depicted Internet tasks which had to be solved orally. The tasks were categorized according to Kellar's Web information task classification [7]:

- Information Seeking: e.g., fact finding
- Information Exchange:
  - Transactions: e.g., hotel booking
  - Communications: e.g., sending a Facebook message

The analysis of the speech data revealed that a natural communication style occurred most frequently in information seeking tasks. As for information exchange tasks subjects used natural communication and command-based speaking style equally. This result is valid for transaction and communication tasks. Because of the equal frequency of occurrence we have to examine which speech dialog strategy is the most suitable for performing information exchange tasks before starting to develop a SDS.

This paper reports from work in progress in which different in-car SDS are compared. The SDS are based on different speech dialog strategies, a command-based and a conversational dialog, which will be evaluated on usability and distraction. The systems have been developed for German users and allow for performing an Internet activity for information exchange (using the example of a hotel booking service) by speech. As common in in-car SDS the speech interaction is supported by a graphical user interface (GUI). Different GUIs are designed in order to support the respective dialog strategy and to raise the level of naturalness. This research is conducted within the scope of the EU FP7 funding project GetHomeSafe.<sup>1</sup>

The following paper is structured as follows: Sect. 2.2 gives an overview on previous studies on this research topic. In Sect. 2.3 the functionality of the hotel booking service is explained. Section 2.4 presents the different human-machine interaction (HMI) concepts which are developed within this research work. Here, the different speech dialog strategies and the different GUI concepts are explained. Section 2.5 describes the planned experiments to be conducted in the near future in order to evaluate the different HMI concepts and, finally, conclusions are drawn.

---

<sup>1</sup><http://www.gethomesafe-fp7.eu>.

## 2.2 Related Work

First studies on the evaluation of dialog strategies have been conducted by Devillers et al. [2] who compare two SDS allowing the user for retrieving touristic information. One dialog strategy guides the user via system suggestions, the other does not. The evaluated dialog strategies comprise the fundamental ideas our command-based and conversational dialog strategy consist of (which are explained in detail in Sect. 2.4). By applying qualitative and quantitative criteria, they conclude that user guidance is suitable for novices and appreciated by all kind of users. However, there was no GUI involved and the speech interaction was performed as primary task. Considering the driving use case other results may be achieved since the primary task is driving.

In the TALK project [10] a command-based speech dialog has been compared to a conversational dialog in the automotive environment. Here, the primary task was driving and as secondary task the driver had to control the in-car mp3-player by speech. The same GUI was used for both dialog strategies. In the field test the subjects had to use the different SDS while driving. Although the conversational dialog was more efficient, the command-based dialog was more appreciated by the subjects. According to Mutschler et al. the high error rate of the conversational strategy was the reason for the higher acceptance of the command-based dialog. The driving performance has been measured with the help of different driving data (e.g., lane keeping). There were no significant differences revealed in the driving performance when using the different SDS.

The speech recognizer quality has improved enormously within the last 5 years. Therefore, the weak speech recognition performance of Mutschler et al.'s conversational dialog may be nowadays less significant. Furthermore, the use of the same GUI for different dialog strategies could have additionally influenced the result. The GUI should be adapted to the particular dialog strategy in order to best benefit from the advantages of the respective strategy and to allow for a comparison of optimal systems. When evaluating the driving performance the averting of the driver's gaze towards the GUI has not been taken in consideration. A glance on the head unit screen could be dangerous if a cyclist would cross the street. The visual distraction can cause accidents which could not be detected with the current performance measurements. Depending on the dialog strategy the visual distraction differs which has to be examined and needs to be compared.

## 2.3 Functionality of the Hotel Booking Service

The chosen use case for the design of the HMI concepts is booking a hotel by speech while driving. For this purpose, the online hotel booking service HRS<sup>2</sup> has been

---

<sup>2</sup><http://www.hrs.com>.

linked to the existing speech dialog framework. The interface and the functionality of the HRS service are briefly described.

The Web service has been linked via the provided SOAP interface into the existing framework. When sending SOAP XML requests via the interface, the service responds with the requested information encapsulated in a SOAP XML message.

The hotel service HRS allows for various hotel search functions. After having input several required parameters (e.g., location, arrival date), the service delivers a list of hotels which match the search criteria. Additionally, there is the opportunity to enter optional parameters (e.g., price range) to refine the search. The user is able to sort the result list in a certain order or filter according to desired hotel facilities (e.g., swimming pool, parking). The service offers a detailed description of each hotel. After having selected a certain hotel, it can finally be booked.

The mentioned functions have been taken into consideration for the different HMI concepts. Each concept has been designed to allow for parameter input, result list presentation, filtering, and sorting. When using the SDS prototypes, the retrieved hotel data correspond to the currently available hotel information, the booking is only simulated. HRS offers many more functions; however, these functions have not been considered when designing the HMI concepts since they would not have been of additional use to compare the different concepts and, therefore, they were not implemented.

## **2.4 HMI Concepts**

In this section the various HMI concepts are described. First, the different dialog strategies including sample dialogs are presented. Afterwards the GUI concepts, which have been designed in order to support the speech dialog, are described with the aid of screenshots.

### ***2.4.1 Dialog Strategy Design***

Two different dialog strategies, a command-based and a conversational dialog strategy, have been designed, and prototypes have been implemented for the later evaluation.

The following technical SDS features were integrated in both prototypes: in order to speak to the system the driver has to press a push-to-activate (PTA) button. Furthermore, the driver is able to interrupt the system while prompting the user (“barge in”). State-of-the-art in-car SDS use “teleprompters” to inform the driver visually about possible commands. However, the use of “teleprompters” raises too much visual attention on the head unit screen. Therefore, the user is only informed audibly about possible commands.

The developed speech dialog prototypes have been specified for German language. However, the sample dialogs given in this section are written in English for better understanding. The characteristic of each strategy and how they differ are described in the following. When designing the different dialog strategies, we particularly focused our attention on the dialog initiative, the possibility to enter multiple input parameters, and the acoustic feedback.

#### 2.4.1.1 Command-Based Dialog Strategy

The dialog behavior of the command-based dialog strategy corresponds to the voice-control which can be found in current state-of-the-art in-car SDS. By calling explicit speech commands, the speech dialog is initiated and the requested information is delivered or the demanded task is executed. There are several synonyms available for each command. By using implicit feedback in the voice prompts, the driver is informed about what the system has understood. After the first command the user is guided by the system and executes the steps which are suggested and displayed by the system. The GUI supports the speech dialog by showing the “speakable” commands as widgets on the screen (see Sect. 2.4.2). A sample dialog is illustrated in the following:

Driver: *Book a hotel.*  
System: *Where would you like to book a hotel?*  
Driver: *In Berlin.*  
System: *When do you want to arrive in Berlin?*  
Driver: *Tomorrow.*  
System: *How long would you like to stay in Berlin?*  
Driver: *Until the day after tomorrow.*

When the parameters have been input, HRS is called to retrieve the list of hotels. The user can then continue the interaction by calling certain commands.

#### 2.4.1.2 Conversational Dialog Strategy

In the conversational dialog strategy, the dialog initiative switches during the speech interaction. The driver is able to speak whole sentences where multiple parameters can be set within one single utterance. Thereby, the dialog can run more naturally, be flexible and efficient. The driver is informed about what the system has understood by using implicit feedback. If the driver has set multiple parameters in his utterance, the system does not implicitly repeat all parameters as the system response would be too long. Therefore, the system repeats only the contextually most important parameter. The GUI does not present the “speakable” commands on the screen. In order to indicate the possible functions, icons are displayed (see Sect. 2.4.2). A sample dialog is presented in the following:

Driver: *I would like to book a hotel in Berlin.*  
System: *When do you arrive in Berlin?*  
Driver: *I arrive tomorrow and leave the day after tomorrow.*

As illustrated in the example, the driver can already indicate some input parameters when addressing the system for the first time. The system checks which input parameter are missing in order to send a request to HRS. The system prompts the user and collects the missing information. Although the system asks for only one parameter, the user is able to give more or other information than requested.

When the parameters have been input, HRS is called to retrieve the list of hotels. The user can now continue the interaction by speaking freely and without having to call certain commands.

### 2.4.1.3 Comparison of Dialog Strategies

The TRINDI ticklist from Bohlin et al. [1], which characterizes the dialog behavior of a SDS with the help of 12 Yes-No questions, gives a good overview of the implemented dialog features. Both of the SDS prototypes have been developed and differentiated corresponding to this list. The filled out TRINDI ticklist for both dialog strategies is illustrated in Table 2.1.

In this research work the most important dialog features which allow for a differentiation of both dialog strategies have been realized so far. Concerning the dialog design of the conversational dialog, we set a high value on the flexibility to input parameters by speech (e.g., Q2, Q3, Q12). Dialog features which are no beneficial characteristic of one of the dialog strategies and which do not reveal differences in the evaluation are left out to lower the development effort (e.g., Q5, Q6, Q8). Impact of the environment on the speech interaction is not in focus of this research (Q8). The dialog flow of a hotel booking dialog is linear and does not allow for context-relevant branches whereby Q11 becomes superfluous.

## 2.4.2 GUI Design

The different GUIs have been designed in order to support the speech dialog strategies the most and to raise the level of naturalness in the interaction. The different GUIs have been customized corresponding to the dialog strategies only as much as necessary since an objective comparison is targeted. When designing the screens we followed the international standardized AAM guidelines [3] which determine the minimum font sizes, the maximum numbers of widgets, etc., in order to minimize distraction. In the following the general differences of the different GUI concepts are described with the aid of screenshots.

**Table 2.1** Characterization of speech dialog strategies on the basis of the TRINDI ticklist

	Command-based dialog	Conversational dialog
Q1:	<i>Is utterance interpretation sensitive to context?</i>	
	✓	✓
Q2:	<i>Can the system deal with answers to questions that give more information than was requested?</i>	
	✗	✓
Q3:	<i>Can the system deal with answers to questions that give different information than was actually requested?</i>	
	✗	✓
Q4:	<i>Can the system deal with answers to questions that give less information than was requested?</i>	
	✓	✓
Q5:	<i>Can the system deal with ambiguous designators?</i>	
	✗	✗
Q6:	<i>Can the system deal with negatively specified information?</i>	
	✗	✗
Q7:	<i>Can the system deal with no answer to a question at all?</i>	
	✓	✓
Q8:	<i>Can the system deal with noisy input?</i>	
	Not in scope of the research work	
Q9:	<i>Can the system deal with “help” sub-dialogs initiated by the user?</i>	
	✗	✗
Q10:	<i>Can the system deal with “non-help” sub-dialogs initiated by the user?</i>	
	✗	✓
Q11:	<i>Does the system only ask appropriate follow-up questions?</i>	
	No relevant dialog step in existent hotel booking dialog	
Q12:	<i>Can the system deal with inconsistent information?</i>	
	✗	✗

### 2.4.2.1 Command-Based Dialog GUI

In the command-based dialog strategy, the driver uses commands to speak to the system. In order to give the driver an understanding of the “speakable” commands, the speech dialog is supported by the GUI. For that reason the currently possible speech commands are displayed on the screen at all times, which may lead to a high visual distraction. Hence, in automotive terms the command-based speech dialog strategy is also called “speak-what-you-see” strategy.

Figure 2.1 illustrates the main screen of the hotel booking application at the beginning of the hotel booking dialog. Here, the first input parameter “destination” (“Ziel” in German) has been set by the user after being requested by the system. Afterwards the user is guided step-by-step by the system. When the driver has given the requested information, a new widget appears on the screen and the system asks the driver for the corresponding input.

**Fig. 2.1** Main screen of the command-based dialog while parameter input



**Fig. 2.2** Main screen of the command-based dialog after parameter input



When all the parameters are set and the hotel service has returned the list of hotels, the list of filters is displayed and the possible commands for changing the input parameters (“Suche ändern”), setting the hotel facilities (“Ausstattung”), sorting the result list (“Sortieren”), and presenting the result list (“Liste”) become visible in the sub-function line (see Fig. 2.2). The active GUI state after receiving the list of hotels is the “Suche ändern” screen where the search parameters, which are presented in the main area of the main screen (e.g., “Ziel” or “Ankunft”), can be changed. However, the driver has several possibilities to proceed with the speech dialog by calling the other commands displayed in the sub-function line. By calling the command “Ausstattung” (or synonyms of the command) the filter sub-dialog is triggered and the hotel facility screen is displayed (see Fig. 2.3). For the presentation and the sorting of the result list there are further similar screens.

#### 2.4.2.2 Conversational Dialog GUI

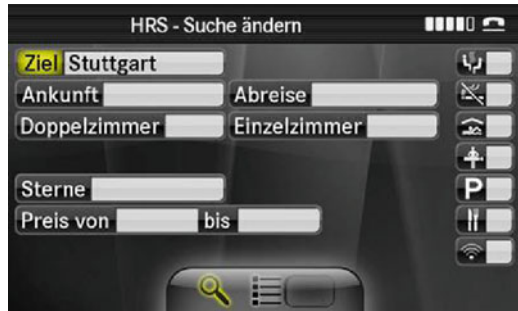
In the conversational dialog strategy, the driver can speak freely and does not have to call certain commands. There is no need to give the driver a visual feedback of the currently “speakable” commands whereby the visual distraction may be lowered. For that reason, the content on the head unit screen does not have to indicate the possible options to proceed with the speech dialog. The sub-function line which was used to indicate the available commands is replaced by only few symbols which resemble the current GUI state.



**Fig. 2.3** Hotel facilities screen of the command-based dialog



**Fig. 2.4** Main screen of the conversational dialog at the beginning of the interaction



**Fig. 2.5** Main screen of the conversational dialog after parameter input



Figure 2.4 shows the main screen at the beginning of the speech interaction. The user is able to input several parameters at once. He is even allowed to already set the hotel facility filters.

After having input all required parameters (and optional parameters or filters) the system calls the HRS service and retrieves a list of hotels (see Fig. 2.5). In this GUI state the driver is able to change the search parameters, change the hotel facility filters, or sort the list by speech. There are no additional screens for presenting the available filters or for the list sorting options. The alterations evoked by speech become only visible on the main screen by changing the information displayed.

**Fig. 2.6** Main screen of the conversational dialog with avatar after parameter input



The symbols on the bottom of the screen resemble the GUI states for parameter input/changes and the result list. The design of the result list screen is the same as the one concerning the command-based strategy.

### 2.4.2.3 Conversational Dialog GUI with Avatar

The goal of using an avatar is to raise the naturalness of the HMI. By expressing gestures and mimics, the avatar contributes to a more human interaction. When seeing a human character on the screen, the driver might tend to speak more naturally, as if he would talk to a human being. This might have a positive effect on speech dialog quality and user acceptance. However, the user might be more distracted by a human character on the screen. So far, those positive and negative effects of an SDS with avatar while driving have not been examined.

The GUI concept with avatar is based on the conversational dialog GUI. A virtual character designed and developed by Charamel<sup>3</sup> is integrated. The avatar overlays the background illustrated in Figs. 2.4 and 2.5 but does not cover the widgets which are currently important for the speech dialog (see Fig. 2.6).

When the driver is driving without interacting with the SDS there is no avatar visible on the screen. The human agent appears when the speech dialog is initiated. When the speech dialog is finished, the avatar disappears again. In this way, the visual distraction is lowered and the driver knows when he is allowed to speak to the system. The avatar makes certain gestures to give the SDS some human character. For example, when the system asks for inputting the destination, the avatar points at the destination widget on the screen. When the user browses the hotel result list, the avatar makes a swipe gesture to support the scrolling in the list.

<sup>3</sup>[www.charamel.de](http://www.charamel.de).

## 2.5 Evaluation

The speech-based HMI concepts that were introduced above will be evaluated with the help of formative user studies in order to test usability and driver distraction. Based on the results of the experiments, the best HMI concept will be employed in the GetHomeSafe system and will be further improved.

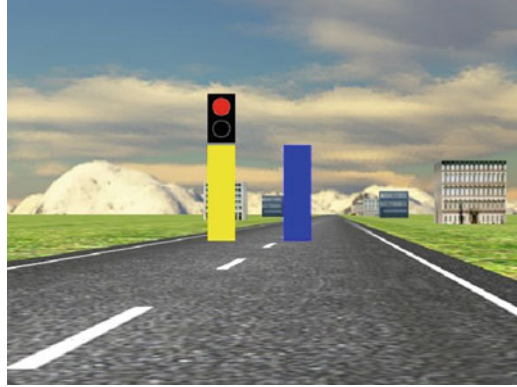
As a first step, a small number of subjects will test the different speech dialog strategies while performing the standard lane change task (LCT) [8]. With the help of this rather explorative test, we will prove if the actual user expectancies are met and potential system shortcomings, such as grammar deficiencies, can be corrected. As a next step, we plan to evaluate the mentioned HMI concepts by conducting a more substantial user study in the driving simulator at DFKI's "future lab" (see Fig. 2.7). We will employ the OpenDS open source driving simulation which is being developed and improved within the scope of the EU research project GetHomeSafe. In this study, the command-based dialog strategy used as reference system is tested only with GUI, whereas the conversational dialog will be presented without GUI, with GUI, and with GUI including the avatar.

As a primary driving task in the second study, we will use the ConTRe (continuous tracking and reaction) [9] task which complements the de facto standard LCT including higher sensitivity and a more flexible driving task duration without restart interruptions. Another requirement for our evaluation is a more fine-grained assessment of driver distraction, in terms of temporal resolution of performance metrics. In LCT, drivers are only once in a while directed to change the lanes (even with announcement) by conducting a rather unnatural abrupt maneuver, combined with simple lane keeping on a straight road in between. But real driving mostly demands a rather continuous adjustment of steering angle and speed without announcements, when the next demand will occur exactly and to which extend a reaction will be necessary. In order to receive more detailed results about the two diverse dialog strategies, we use a task that rather resembles continuous driving, like a car following task. Furthermore, we prefer an absolute ground truth of perfect



**Fig. 2.7** DFKI driving simulator setup

**Fig. 2.8** Screenshot of the ConTRe task as the first modular extension of the OpenDS simulation component



behavior for the performance metric, whereas the LCT is based on an ideal line as a generated, normative model. Another intended advantage of the ConTRe task over the LCT and also many other standard tasks is the possibility to explicitly address mental demand via an event detection task. Effects of cognitive load should be revealed above all by the achieved reaction times. Therefore, an additional discrete task was implemented as longitudinal control (gas and brake). This task should be accomplished in addition to the continuous adjustment of steering wheel angles for lateral control.

The driver's primary task in the simulator is comprised of actions required for normal driving: turning the steering wheel, as well as operating the brake and acceleration pedals. System feedback, however, differs from normal driving. In the ConTRe task, the car moves on its own with a constant speed through a predefined route on a unidirectional straight road consisting of two lanes. Turning the steering wheel moves the car laterally but no further than the edge of the carriageway. Additionally, steering manipulates a moving blue bar, which is rendered in front of the car (see Fig. 2.8). On the road ahead, the driver perceives this blue bar and another yellow bar, both moving continuously at a constant longitudinal distance in front of the car. The yellow one is called the reference bar, as it moves autonomously within the roadsides according to an algorithm. The driver controls the lateral position of the blue bar by turning the steering wheel, trying to keep it overlapping with the reference bar as well as possible. A distance metric between the reference bar and the controllable bar is recorded continuously. Effectively, on an abstract level this corresponds to a task where the user has to follow a curvy road or the exact lateral position of a lead vehicle, although correct task performance is indicated more obviously and therefore leads to less user-dependent variability.

In addition to the steering task, common gas and brake reactions are also required once in a while. However, operating the acceleration or brake pedal does not have any effect on vehicle speed. There is a traffic light placed on top of the reference cylinder containing two lights: the lower one can be lighted green, whereas the top light shines red when it is switched on. Only one of these lights appears once in a while. The red light requires an immediate brake reaction with the brake pedal,

whereas green indicates that an immediate acceleration with the gas pedal should be performed. As soon as the driver reacts correctly, the light turns off (see Fig. 2.8). Reaction time as well as accuracy can be assessed.

Besides measuring driver distraction via performance metrics, we will assess subjective mental workload with the help of the DALI questionnaire [11] after each system condition. Eye tracking will be used for gaze-based distraction evaluation, including average and maximum glance duration for the different GUI variants. A qualitative assessment of the dialog strategies will be performed using the PARADISE framework [13], which appraises overall dialog quality by means of several interaction criteria (e.g., success rate, number of interaction steps). The SASSI questionnaire [6] will be used to survey subjective usability evaluation of the speech dialog variants. Previous knowledge of participants on SDS will be assessed in the very beginning as part of a biographic questionnaire.

Overall, we expect better usability evaluation for the conversational dialog conditions compared with the command-based condition. For the conversational dialog conditions, we do not expect large differences regarding usability when comparing the conditions with GUI with the conditions without GUI. However, for this comparison we expect the GUI to cause more driver distraction in terms of glances onto the GUI screen and in terms of decreased driving performance. For these metrics we expect the command-based GUI variant to perform worse than the conversational GUI. Furthermore, we expect to find longer task completion times for the command-based dialogs. If increased task duration occurs on the same level of performance decrease, the condition with shorter task duration should be chosen. When using the avatar we expect positive effects on usability. However, we expect the GUI with avatar to cause more driver distraction than the normal conversational GUI. The presented experimental investigation will help us to decide about the most preferable dialog strategy and about what kind of GUI should be employed.

## 2.6 Conclusions

This paper reports from work in progress in which different in-car speech-based HMI concepts are compared. For each concept a prototype which allows for an online hotel booking has been developed.

The described HMI concepts are based on different dialog strategies which include speech as main input and output modality. The speech dialog is supported by a GUI which is adapted to the respective speech dialog strategy. The first HMI concept is based on a command-based dialog strategy where the driver is able to start the speech dialog by single commands and is led step-by-step by the system afterwards. The available commands are displayed on the head unit screen. The second dialog strategy, the conversational dialog, allows the driver to speak with entire sentences as if he would talk to a human being. Thereby, multiple parameters can be input at once and the dialog initiative switches frequently. Two different GUI design concepts were targeted to support the conversational dialog and to raise the level of naturalness. The first concept does not display the commands anymore but

uses icons to suggest possible functions of the system to the driver. Based on the first GUI concept, the second concept contributes to a more conversational interaction by displaying additionally a humanlike character on the screen.

With the aid of the developed prototypes the different HMI concepts will be evaluated on usability and driving performance. The driving simulator experiments will be performed at DFKI in Saarbrücken. Based on the results of the experiments, the best HMI concept will be employed in the GetHomeSafe system and will be further improved.

**Acknowledgements** The research work described in this paper is performed in the context of the GetHomeSafe project which is conducted within the scope of the Seventh Framework Program of the European Commission. We would like to thank the European Commission for funding the GetHomeSafe project.

## References

1. Bohlin, P., Bos, J., Larsson, S., Lewin, I., Matheson, C., Milward, D.: Survey of existing interactive systems. Deliverable 1.3, TRINDI (1999)
2. Devillers, L., Bonneau-Maynard, H.: Evaluation of dialog strategies for a tourist information retrieval system. In: Proceedings of International Conference on Spoken Language Processing, pp. 1187–1190 (1998)
3. Driver Focus-Telematics Working Group: Statement of Principles, Criteria and Verification Procedures on Driver Interactions with Advanced In-Vehicle Information and Communication Systems. Alliance of Automotive Manufacturers, Washington (2002)
4. Governors Highway Safety Association: Distracted driving: What research shows and what states can do. Technical report, U.S. Department of Transportation, 2011
5. Hofmann, H., Ehrlich, U., Berton, A., Minker, W.: Speech interaction with the Internet - a user study. In: Proceedings of Intelligent Environments, Guanajuato, Mexico (2012)
6. Hone, K.S., Graham, R.: Subjective assessment of speech-system interface usability. In: Proceedings of Eurospeech, pp. 2083–2086 (2001)
7. Kellar, M.: An examination of user behaviour during web information tasks. Ph.D. thesis, Dalhousie University, Halifax, Canada (2007)
8. Mattes, S.: The lane-change-task as a tool for driver distraction evaluation. Proceedings of IGfA, pp. 1–30 (2003)
9. Moniri, M., Mahr, A., Math, R., Feld, M., Müller, C.: The conTRe (continuous tracking and reaction) task: A flexible approach for measuring driver distraction with high sensitivity. In: Adjunct Proceedings of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, pp. 88–91 (2012)
10. Mutschler, H., Steffens, F., Korthauer, A.: Final report on multimodal experiments - part 1: Evaluation of the sammie system. d6.4. talk public deliverables. Technical Report (2007)
11. Pausie, A.: Evaluating driver mental workload using the driving activity load index (DALI). In: Proceedings of European Conference on Human Interface Design for Intelligent Transport Systems, pp. 67–77 (2008)
12. Peissner, M., Doebler, V., Metze, F.: Can voice interaction help reducing the level of distraction and prevent accidents? Meta-study on driver distraction and voice interaction. Technical Report, Fraunhofer-Institute for Industrial Engineering (IAO) and Carnegie Mellon University, 2011
13. Walker, M.A., Litman, D.J., Kamm, C.A., Kamm, A.A., Abella, A.: PARADISE: A framework for evaluating spoken dialogue agents. In: Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics, pp. 271–280 (1997)



<http://www.springer.com/978-1-4614-8279-6>

Natural Interaction with Robots, Knowbots and Smartphones  
Putting Spoken Dialog Systems into Practice

Mariani, J.; Rosset, S.; Garnier-Rizet, M.; Devillers, L. (Eds.)

2014, XVI, 397 p. 132 illus., 73 illus. in color., Hardcover

ISBN: 978-1-4614-8279-6