

MIMI: A Multimodal and Adaptive Interface for an In-Car Communication System

Patrick Tchankue, Janet Wesson and Dieter Vogts

Department of Computing Sciences

Nelson Mandela Metropolitan University, P. O. Box 77000, Port Elizabeth 6031

Tel: +27 41 5042323, Fax: +27 41 5042831

email: {Patrick.TchankueSielinou, Janet.Wesson, Dieter.Vogts}@nmmu.ac.za

Abstract- The number of cars provided with in-car communication systems has increased noticeably. Research has shown that using a mobile phone whilst driving has usability and safety issues. In order to minimise these issues, user interfaces that promote the usage of the hands and eyes solely for the driving task have been proposed. This paper discusses the design of MIMI (Multimodal Interface for Mobile Infocommunication), a prototype multimodal in-car communication system using a speech interface, supplemented with steering wheel button input. The design of MIMI is discussed, together with its architecture, including the dialogue manager, the adaptive module and the workload manager. The adaptive module contains a workload manager that helps MIMI to present information to the driver such that driver distraction is reduced.

Index Terms — Multimodal Interfaces, Adaptive Interfaces, In-Car Communication Systems, Workload Manager, Neural Networks.

I. INTRODUCTION

An increasing number of new in-car systems have been proposed to enhance comfort, safety and driving performance in modern cars. These systems, however, can decrease the driver's driving capacity, making the driving situation potentially hazardous. It has been shown that almost half of vehicle accidents are caused by driver distraction [1], where the driver is distracted by a lack of concentration, an outside event or the use of in-car applications such as navigation or in-car communication systems (ICCS). In many countries, including South Africa [2], it is illegal to use a mobile phone whilst driving unless the car is provided with a hands-free system through which the driver may interact with the mobile phone.

Driver distraction is multidimensional and has been classified into three types, namely visual distraction, biomechanical distraction and cognitive distraction. Visual distraction occurs when drivers remove their gaze from the road for some reason, biomechanical distraction when drivers use their hands to manipulate an object and cognitive distraction occurs when drivers think about something other than the task of driving. Several studies have shown that usage of the speech channel can help to reduce the distraction experienced by the driver when engaged in a

secondary task. Interfaces that promote eyes-free and hands-free modality (e.g. gesture, eye-gaze and speech) can help the driver to focus on the road even while interacting with a mobile phone.

Communication using the speech channel places less of a cognitive load on the driver, but can be ambiguous. Use of an additional input channel, such as steering wheel button input can reduce this ambiguity. Using simple, clear dialogue flow, with appropriate feedback and presenting information at a less distracting moment may also reduce the cognitive load placed on the driver.

Usability and safety issues can occur when using current ICCS. This paper aims to introduce and discuss the design of MIMI as a possible solution to the above issues.

This paper is organised as follows: background information about ICCS, multimodal and adaptive interfaces is discussed in Section 2. Section 3 discusses the design of MIMI, with the first section presenting MIMI's architecture and subsequent sections describing the individual components in more detail. The adaptive module and workload module are discussed in Section 4, while Section 5 details the implementation of MIMI. Section 6 contains a discussion of the possible benefits that ICCS can gain through the proposed architecture and Section 7 concludes this paper.

II. RELATED WORK

Previous studies of ICCS, driver distraction, multimodal interfaces and adaptive interfaces were considered in the development of MIMI and are briefly reviewed below.

A. In-Car Communication Systems (ICCS)

ICCS are systems that can help drivers to interact with a Bluetooth-enabled phone paired with the system and perform typical tasks such as recalling names in the address book [3]. Existing ICCS such as SYNC, iDrive and COMAND, not only manage the communication system, but also control the radio, provide vehicle information (via the CAN network), and include a navigation system.

It has been found that drivers are more likely to be involved in vehicle accidents when using mobile phones or high visual demand devices [4], which means that using an ICCS can cause distraction.

B. Driver Distraction

According to the literature, driver distraction happens when the driver's attention is, voluntarily or involuntarily,

diverted away from the driving task by an event or an object to the extent that the driver is no longer able to perform the driving task adequately or safely [5, 6].

Several types of visual distraction exist [5] when using ICCS while driving:

- The driver neglects to look at the road and instead focuses on another visual target, such as an in-car route navigation system, for an extended period of time.
- The driver loses his visual “attentiveness”. This distraction is also known as “looked, but did not see” [7] and can happen if the driver tries to figure out who he is talking to, after having accepted the call without feedback.

Biomechanical distraction occurs when a driver removes one or both hands from the steering wheel to physically manipulate an object instead of focusing on the road.

Dialling has been found to affect driving performance, specifically the lateral vehicle control, due to biomechanical interference with steering [8].

Text messaging is also a major source of manual distraction. A recent study in Monash University Accident Research Centre shows that sending an SMS is more distracting than simply talking on a mobile phone [9]. Cognitive distraction includes any thoughts that absorb drivers’ attention to the point where they are unable to navigate safely and reaction times are reduced [10].

C. Multimodal Interfaces

According to Oviatt [11], multimodal interfaces are interfaces that process two or more combined user input channels in a coordinated manner. Multimodal interfaces have been widely adopted because they offer several benefits:

- *Flexibility*: With several methods of interaction, users have a choice of input and/or output modalities. This might be necessary, as in the case of a user who is unable to communicate via a particular modality such as vision, speech or gesture (e.g. deafness or paraplegia).
- *Efficiency*: Certain tasks can be performed more quickly with appropriate input or output devices (e.g. dialling a phone number using voice commands instead of manually entering the phone number). The use of additional communication channels can speed up the task completion by 10% [11].
- *Error-handling or cross-modal synergy*: Errors in one mode (e.g. imprecision, ambiguity or incorrect speech input) can often be corrected by processing and integrating synchronous input from another modality (e.g. manual input). Error handling and reliability can be improved as it has been found that users make 36% fewer errors with a multimodal interface than with a unimodal interface [11].
- *User experience enhancement*: Multimodal interfaces can help to decrease the user’s stress and increase the user’s motivation [11].

D. Adaptive User Interfaces

The goal of an adaptive user interface is to adapt the user interface to a specific user, provide feedback about the user’s knowledge and predict the user’s future behaviour such as answers, goals, preferences and actions [12]. An adaptive interface is typically implemented using an

adaptive module comprising several models, including a user model, a task model, a context model and an adaptive engine.

III. ARCHITECTURE OF MIMI

MIMI was built using a pipelined architecture. Such architectures have been successfully applied in several multimodal and spoken dialogue projects such as Microsoft’s MIPAD [13] and in-car multimodal dialogue systems such as SAMMIE (Saarbrücken, Multimodal Mp3 Information Extractor).

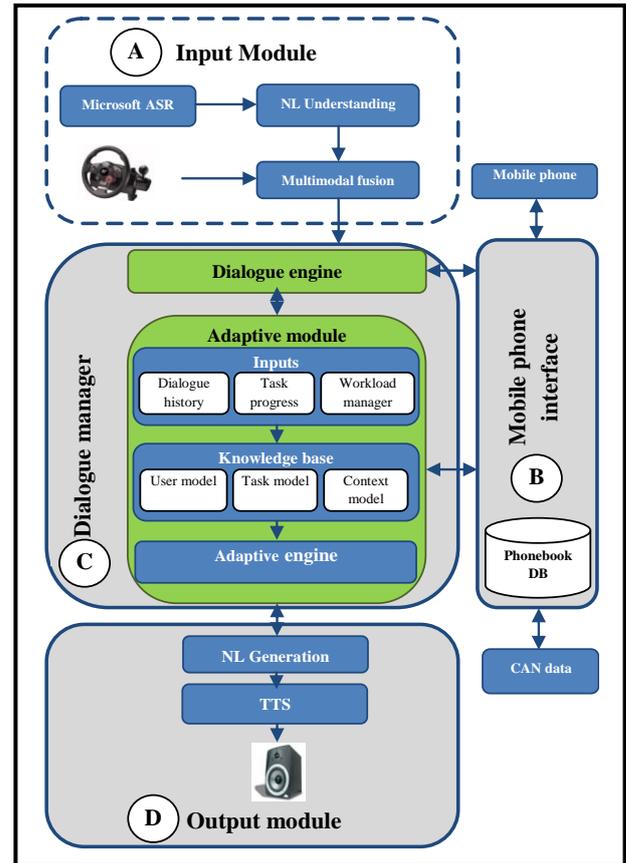


Figure 1: General Architecture of MIMI

Figure 1 illustrates the general architecture of MIMI which is based on the SAMMIE architecture. The main differences between MIMI and SAMMIE lie in the Output Module (D), the Mobile Phone Interface (B) and the Dialogue Manager (C). MIMI’s output module is voice-based: only speech is used to provide feedback to the driver, as any kind of visual display can require the driver to look away from the road. The mobile phone interface does not handle infotainment tasks such as playing Mp3 tracks, so only communication-related tasks are taken into account. An adaptive module is included in MIMI’s dialogue manager (DM) in order to take certain contextual variables into account when interacting with the driver.

MIMI is a mixed-initiative dialogue system in that an action may be initiated by either the driver or by MIMI, two examples of which are shown in Table 1. In the first scenario (Table 1a), MIMI detects an incoming text message by polling the mobile phone interface, which creates and sends a frame to the DM. The dialogue engine requests the current distraction level from the adaptive module. If the distraction level is too high, the output events are stored in a

queue and are only sent to the output module when the distraction level has dropped to an acceptable level. In the second scenario (Table 1b), the driver gives a spoken command and the output module creates the appropriate frame, which is sent to the DM. From there, the frame is processed in a similar manner to the first scenario.

(a) Mobile phone event: incoming text message	
1	Event: Incoming text message
2	MIMI: acknowledge the incoming text message and retrieve the phone number, the date, the time and the actual message.
3	MIMI: uses the previous information to build a frame that the DM will use.
4	MIMI: the DM, assisted by the adaptive module, decides to notify the driver if the current distraction level is acceptable.
(b) Calling a contact	
1	User: Call John Doe
2	MIMI: Do you want to call John Doe on 0771112222. Say 'Yes' or 'No'
3	User: Yes
4	MIMI: Calling John Doe...

Table 1: Mixed-initiated Dialogues of MIMI

MIMI's architecture is comprised of four interacting modules, namely:

A. The Input Module

This module can accept input on two channels, namely the audio channel and the manual controller channel. Speech input is accepted by the Automatic Speech Recogniser (ASR) and sent to the Natural Language Understanding (NLU) module, which interprets the driver's spoken commands. Input from the dashboard and steering wheel buttons are accepted on the manual controller channel. The Multimodal Fusion component combines the input from both channels to determine what the user's command is in a reliable and unambiguous manner.

In order to keep track of and efficiently manage inputs, the Multimodal Fusion component uses an *Input Pool* to store data from speech and button inputs. Whenever an input is created, it is inserted at the end of the input pool. Each input has a lifecycle of 4 minutes (duration of a dialogue depends on the task and ranges from 20 to 200 seconds [14]) in the input pool. If an input is not processed before the end of its lifetime, it is removed from the pool.

Once the input has been combined and disambiguated, the input module creates and sends a frame to the Dialogue Manager (DM).

B. The Mobile Phone Interface

The Dialogue Manager (DM) interacts with the mobile phone interface application. The DM may send commands to the mobile phone interface application or receive commands wrapped in a frame in the case of external events, such as incoming calls or text messages, being detected. Contextual variables that are needed to estimate the distraction level of the driver may also be obtained from the mobile device via the mobile phone interface application. A database is used to store contacts, call and text message history.

C. The Dialogue Manager

The DM monitors and manages conversations between MIMI and the driver. A frame-based approach for dialogue management is used in the DM.

The DM consists of two components: a *dialogue engine* and an *adaptive module*. The dialogue engine interacts with the adaptive module in order to obtain the current driver distraction level and adapt the output accordingly.

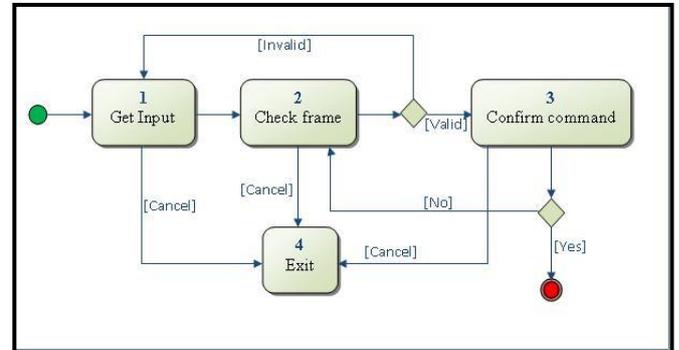


Figure 2: UML Activity Chart Diagram of the DM Engine

The dialogue engine is responsible for the interaction between the input module, the output module and the mobile phone interface.

Figure 2 illustrates the steps performed by the dialogue engine in order to manage the dialogue between the driver and MIMI. The DM receives inputs in the form of a command frame from the driver and validates the frame contents. If the frame is completely filled, it is sent for confirmation by the driver before any actions are performed; otherwise the DM attempt to obtain valid input from the driver once again. If the command frame is not confirmed by the driver, the DM will check the command again; otherwise the command frame is accepted and is sent to the Mobile Phone Interface to be executed. The driver can cancel an operation at each state, in which case the DM will go to the activity *Exit*.

D. The Output Module

The Output Module consists of the Natural Language (NL) Generator and the Text to Speech Module [15]. The NL Generator is responsible for generating appropriate text for the TTS module and performs tasks such as converting text message abbreviations into correct English. The TTS module then outputs the speech to the driver on the speech communication channel.

IV. ADAPTIVE MODULE AND WORKLOAD MANAGER

The adaptive module and the workload manager represent the intelligent components of the MIMI architecture. The adaptive module monitors the driver's actions by collecting inputs (e.g. dialogue history, task progress and distraction level) and updates the knowledge base that is used to predict the driver's future actions. The workload manager is based on a neural network that is trained to estimate the current distraction level.

A. The Adaptive Module

The adaptive module monitors the driver's actions and updates various knowledge bases based on specific rules.

The dialogue history logs all driver actions and system responses. The task progression indicates the completion of a task.

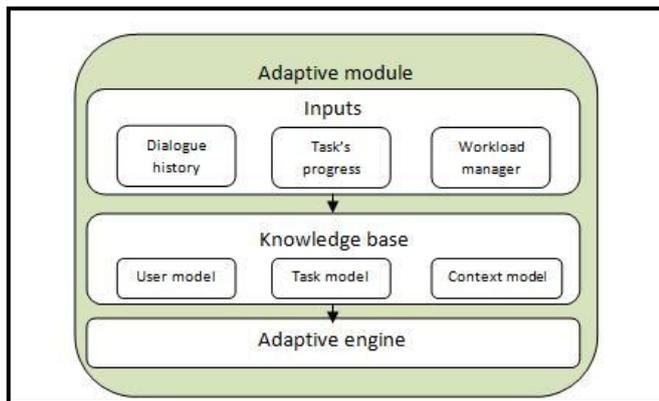


Figure 3: Adaptive Module Architecture

The adaptive architecture illustrated in Figure 3 consists of three sub-components, namely: *input variables*, *knowledge base* and the *adaptive engine*. Input variables consist of the user's actions, the task progress and the estimated driver's cognitive load. These are used as parameters to update all the models in the knowledge base including the context model, which consists of environmental variables (e.g. weather, daytime) and cognitive load estimations. The adaptation engine uses rules to perform the most suitable adaptation of the current task.

The workload manager determines the following levels of distraction: *low*, *mid*, *high* and *extreme*. These distraction levels determine the cognitive load of the driver and are affected by three main factors mapped onto the three models in the knowledge base (Figure 3).

1) User Model

The user model is updated by the dialogue history. It enables the adaptive module to create a user profile which is used to perform some dialogue adaptations based on a specific driver's behaviour.

2) Task Model

The task model is updated by the task progression; each task has a related workload level that determines the system behaviour. For example, sending a text message can be distracting while driving [16, 17]; therefore the distraction level attached to this task is either high or extreme. The consequence is that the driver won't be notified of any other external event whilst sending a text message.

3) Driving Situations

Situations such as lane changing, overtaking, or reversing are considered to be more demanding in terms of attention than other situations. The same applies to certain types of road segments, such as intersections and turns. It is possible to automatically detect such situations by examining contextual variables, such as speed, steering angle, weather, time of day and level of light.

Based on the driving situation and the current task, it is possible to derive a measure of the distraction experienced by the driver and to adapt the behaviour of MIMI accordingly. Examples of task-based adaptation for incoming calls and text messages are:

- 1- Notifying (e.g. the system tells to the driver that there is an incoming call from a contact);

- 2- Delaying/postponing (e.g. the notification of an incoming call or text message is delayed according to the current level of cognitive load) and
- 3- Sending a text message to tell that the driver is not able to respond at the moment (e.g. "Talk to you later").

Table 2 has three examples that illustrate how MIMI handles recognition errors using an adaptive behaviour. In the first example, utterances that are found to be unclear (low confidence rate) can be clarified. In the second example, after the same recognition error occurs several times, the system proposes the most frequent answer that it has stored for this particular case and asks the driver for confirmation. The third example illustrates an automatic adaptation, without requesting any confirmation from the driver, which the system has learnt from the driver after several clarification and grounding stages. This adaptation technique can provide several benefits to drivers [18], including shorter time on the task, a greater completion rate and a reduced cognitive load.

Example 1
User: Call Dieter { <i>Peter</i> recognised, confidence < 40% } System: Are you sure you want to call Peter? Say Yes or No User: No User: Call Dieter { confidence 80% } System: Do you want to call Dieter? Say Yes or No User: Yes System: { interfaces with the mobile phone }
Example 2
User: Call Dieter { <i>Peter</i> recognised, confidence < 40% } System: Do you want to call Dieter? Say Yes or No User: Yes System: { interfaces with the mobile phone }
Example 3
User: Call Dieter { <i>Peter</i> recognised, confidence < 40% } System: { interfaces with the mobile phone }

Table 2: Error Handling through Adaptation

B. The Workload Manager

Several automotive applications of Artificial Neural Networks (ANN) exist to date, including driver workload estimators [19], road-type detectors and traffic predictors [20]. These applications attempt to improve the safety of modern cars through better prediction of driving situations.

The main purpose of MIMI's Workload Manager is to estimate the current distraction level given the current driving context and task. Studies have shown that a large number of vehicle accidents occur due to driving at high speeds. Moreover, critical driving situations such as lane changing, overtaking, reversing and turns are affected by a change in speed. Different weather conditions require different amounts of attention from drivers. In bad weather conditions, drivers are more cognitively loaded than during fine weather conditions and should not be disturbed by a secondary task. At night, drivers are impaired by a lack of light and their visibility is affected; therefore, the time of

day and level of lighting play a role in determining the drivers' cognitive load.

The accuracy of a neural network depends on a number of factors. These factors are: inputs, outputs, the neural network architecture, the training technique and the choice of training data. The following section gives a description of how each of those factors is used in the design of MIMI's workload manager.

1) Inputs:

The speed can range from 0 to 300 kilometres per hour. This can be obtained from the CAN (Controlled Area Network). The steering angle, which can also be obtained through the CAN, ranges from 0 to 360 degrees. The time of day is either day or night, the current date is used to adjust the time of day according to seasons (e.g. days shorter during the winter).

2) Outputs:

The workload manager gives an estimation of the current driver's workload. This workload is ranked either as low, mid, high or extreme.

3) Architecture of the Neural Network:

The workload manager is designed as a classical Multi-Layer Perceptron (MLP) formed by three layers: an input layer, a hidden layer and an output layer [21]. The sigmoid function is used to compute each neuron output either between the input and the hidden layer or between the hidden and the output layer.

4) Training Technique:

A supervised learning technique is used to train the workload manager. The back propagation algorithm [21] is used to compare the differences between the calculated output and the target in order to adapt its weights towards a minimal global error.

5) Training Data:

The effectiveness of a neural network trained by back propagation strongly depends on the reliability of the training set. Possible signs of driver distraction including slowing down in speed, sudden braking, number of collisions or the ability to keep a lane [22] are recorded and associated with the corresponding distraction rate.

V. IMPLEMENTATION

A non-adaptive version of MIMI was implemented in order to identify and remove potential usability issues before implementing the adaptive functionality. This was done in order to prevent any usability issues that could affect the efficiency of the adaptive version of MIMI.

Microsoft Visual Studio 2008 C# was used as the implementation language and the Microsoft Speech API 5.3, shipped with Windows Vista, was used for speech recognition. The 32feet.NET [23] Bluetooth's library was used for discovering and sending commands to Bluetooth devices.

MIMI is designed to be used with six steering wheel buttons namely a Push-to-Talk (PTT) button, a Call button, a Reject button, a SMS button, a Read button and a Pair button (Figure 4).

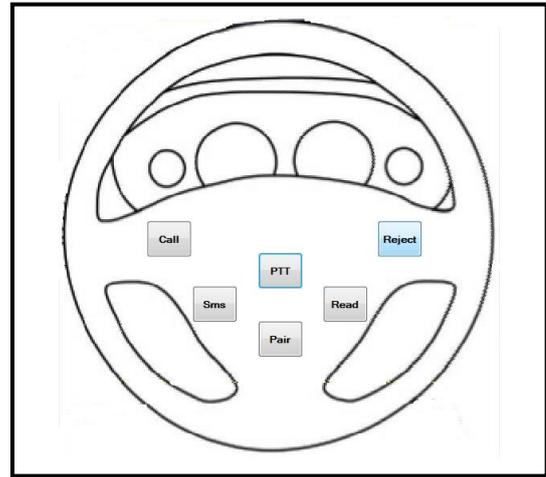


Figure 4: Graphical Simulation of MIMI

Typical text messaging tasks, call tasks and pairing tasks were implemented. Users are able to send text messages by dictating a correct number or selecting a contact in the address book. Then the user has to choose the predefined message that he or she wants to send. Text messages can also be read and abbreviations are converted into simple English. Call tasks can also use dictated numbers and contacts. It is also possible to redial the last outgoing number and call back the last incoming number. The pairing of new Bluetooth-enabled phones is also supported with MIMI.

All commands are transferred to the phone using AT commands through the virtual serial port created by the Bluetooth connection with the device. Several synonyms can be recognised (e.g. dial, call, phone recognised are call commands). These commands can be entered using a combination of speech and buttons.

A usability inspection was conducted by five expert users using Nielsen's heuristics. Usability principles assessed were feedback, error prevention, flexibility and efficiency of use. Two key recommendations were made; namely that the Push-to-Talk button should be used to avoid speech recognition errors and that the predefined messages which can be sent by MIMI as output to the user should be simplified.

VI. DISCUSSION

This paper discussed the design of a safe and usable multimodal interface for an in-car communication system called MIMI. This interface used two input channels, namely speech and steering wheel command buttons and one output channel, namely speech. A pipelined architecture was selected for MIMI because of its simplicity and modularity, which allows designers to add and modify components and modify their interaction with other components. An adaptive module was incorporated into MIMI in order to reduce the drivers' workload and distraction.

MIMI can adapt according to the driver's profile, the current situation and the task status. Adaptive behaviours only affect tasks (incoming calls and text messages) which can be postponed or cancelled according to the current level of distraction. The current level of distraction was determined using a neural network in order to make the system more robust in the presence of noisy input data which may be given by car's sensors through the CAN.

VII. CONCLUSION & FUTURE WORK

The use of in-car communications systems (ICCS) can cause distraction when used whilst driving. This paper has proposed a new multimodal and adaptive architecture, called MIMI, which aims to improve the usability as well as the safety of using an ICCS. Algorithms used to manage the interaction between the driver and MIMI were discussed and their choice motivated in terms of the requirements of an ICCS. The driver's distraction level was determined using an artificial neural network based on variables such as steering angle, speed, weather and time of day. The design of MIMI was partially validated by implementing a non-adaptive version of MIMI which included limited functionality, such as making calls and sending text messages.

Future work will include implementing an adaptive version of MIMI and comparing its usability and usefulness with the non-adaptive version using the driving simulator Lane Change Test (LCT). This study will aim to determine whether the use of a multimodal interface incorporating adaptive techniques can improve system usability and reduce driver distraction.

VIII. REFERENCES

- [1] P. Green, "Crashes induced by driver information systems and what can be done to reduce them," presented at SAE, 2000.
- [2] S. O. Mohammed and F. J. J. Labuschagne, "Can Draconian law enforcement solve the South African road safety crisis?" presented at 27th Annual Southern African Transport Conference, 2008.
- [3] I. Tashev, M. Seltzer, Y.-C. Ju, Y.-Y. Wang, and A. Acero, "Commute UX: Voice Enabled In-car Infotainment System," presented at Mobile HCI '09, Workshop on Speech in Mobile and Pervasive Environments SiMPE, Bonn, Germany, 2009.
- [4] M. J. Goodman, L. Tijerina, and F. D. Bents, "Using cellular telephones in vehicles: Safe or unsafe?" 1999.
- [5] K. Young, M. Regan, and M. Hammer, "Driver Distraction: A Review of the Literature," Monash University, Accident Research Centre 206, 2003.
- [6] K. Young and M. Regan, "Driver distraction: A review of the literature," *Distracted driving*, 2007.
- [7] H. Ito, H. Uno, B. Atsumi, and M. Akamatsu, "Visual Distraction While Driving - Trends in research and Standardization," *IATSS Research*, vol. 25, 2001.
- [8] L. Tijerina and S. Hetrick, "Analytical Evaluation of the Effectiveness of Minimum Separation Distance and Turn-Signal Onset Rules for Lanes Crash Avoidance System Warning Onset," 1996.
- [9] S. Hosking, K. Young, and M. Regan, "The effects of text messaging on young novice driver performance," 2006.
- [10] "The Mobile Phone Report: A report on the effects of using a hand-held and a hands-free mobile phone on road safety," Direct Line Motor Insurance Croydon, United Kingdom 2002.
- [11] S. L. Oviatt, "Multimodal Interactive Maps: Designing for Human Performance," *Human-Computer Interaction*, vol. 12, pp. 93-129, 1997.
- [12] A. Jameson, "Adaptive Interfaces and Agents," 2003.
- [13] X. Huang, A. Acero, C. Chelba, L. Deng, J. Droppo, D. Duchene, J. Goodman, H.-W. Hon, D. Jacoby, L. Jiang, R. Loynd, M. Mahajan, P. Mau, S. Meredith, S. Mughal, S. Neto, M. Plume, K. Steury, G. Venolia, K. Wang, and Y.-Y. Wang, "MIPAD: A Multimodal Interaction Prototype," presented at International Conference on Acoustics, Speech, and Signal Processing, Salt Lake City, Utah, 2001.
- [14] P. Geutner, L. Arevalo, and J. Breuninger, "VODIS - Voice-Operated Driver Information Systems: Usability Study on Advanced Speech Technologies for Car Environments," presented at 6th International Conference on Spoken Language Processing, Beijing, China, 2000.
- [15] J. C. Stutts, D. W. Reinfurt, L. Staplin, and E. A. Rodgman, "THE ROLE OF DRIVER DISTRACTION IN TRAFFIC CRASHES," AAA Foundation for Traffic Safety, Washington, DC 2001.
- [16] F. A. Drews, H. Yazdani, C. N. Godfrey, J. M. Cooper, and D. L. Strayer, "Text Messaging During Simulated Driving," *Human Factors and Ergonomics Society*, 2009.
- [17] S. Hosking, K. Young, and M. Regan, "The effects of text messaging on young novice driver performance," presented at Distracted driving, Sidney, 2007.
- [18] T. Lavie and J. Meyer, "Benefits and Costs of Adaptive User interfaces," *International Journal of Human Computer Studies*, 2010.
- [19] D. Prokhorov, *Computational Intelligence in Automotive Applications*, 2008.
- [20] T. Paek and R. Pieraccini, "Automating spoken dialogue management design using machine learning: An industry perspective," *Speech Communication*, vol. 50, pp. 716-729, 2008.
- [21] A. P. Angelbrecht, *Computational Intelligence: An Introduction*, 2 ed: John Wiley & Sons, 2007.
- [22] S. T. Iqbal, Y.-C. Ju, and E. Horvitz, "Cars, Calls, and cognition: Investigating Driving and Divided Attention," in *CHI 2010*. Atlanta, Georgia, USA, 2010.
- [23] InTheHand, "32feet.NET," vol. 2010, 2010.

Patrick Tchankue received his undergraduate degree in 2001 from Yaoundé I University in Cameroon and is presently studying towards his MSc degree in Computing Sciences at the Nelson Mandela Metropolitan University. His research interests include interaction design, in-car communication systems, multimodal and adaptive interfaces.