User-friendly Multimodal Services - A MUST for UMTS

Going the Multimodal route: making and evaluating a multimodal tourist guide service

Luis Almeida* (1), Ingunn Amdal (2), Nuno Beires (1), Malek Boualem (3), Lou Boves (4), Els den Os (5), Pascal Filoche (3), Rui Gomes (1), Jan Eikeset Knudsen (2), Knut Kvale (2), John Rugelbak (2), Claude Tallec (3), Narada Warakagoda (2)

(1) Portugal Telecom Inovação, (2) Telenor R&D, (3) France Télécom R&D, (4) University of Nijmegen, (5) Max Planck Institute for Psycholinguistics

Abstract

The success of new mobile services depends on the added value perceived by the user, and on the user-friendliness of these services. Multimodal interfaces allow users to select the most appropriate and suitable input and output modalities for interacting according to usage-context. The aspects involved in the implementation of a service with a multimodal interface and usability issues related with this new kind of interfaces are studied in EURESCOM project MUST (MUltimodal, multilingual information Services for small mobile Terminals). This paper focuses on the technical issues of the implementation of a multimodal speech-centric tourist guide for Paris and on the evaluation performed by usability experts.

1. Introduction

For Telecom Operators it is essential to invoke the widest possible use of their future UMTS services. The problems with the introduction of WAP services have proved that wide usage presupposes that at least two requirements are fulfilled: customers must have the feeling that the service offers more or better functionality than existing alternatives, and the service must have a simple and natural interface. Especially the latter requirement is difficult to fulfil with the interaction capabilities of the small lightweight mobile terminals. Multimodal interfaces may solve many of these usability problems and improve the user-friendliness of the new services on small mobile terminals. With a multimodal interface people can ask questions orally while tapping on the screen, and the system combines and interprets these inputs together. The response from the multimodal terminal is represented with graphics, text and speech. However, the combination of multiple input and output modes in a single session appears to pose new technological and human factors problems of its own. Therefore, the Research departments of three Telecom Operators collaborate with two academic institutes in the two year EURESCOM project MUST - *MUltimodal, multilingual information Services for small mobile Terminals* - that has two main aims:

(1) To obtain knowledge about the issues involved in the implementation of a multimodal application for a small, mobile terminal;

(2) To obtain information about user behaviour when using this multimodal application.

We have implemented a multimodal application using a Compaq iPAQ personal digital assistant (PDA) as the terminal. An interactive tourist guide to Paris was chosen as an example service to focus the development work and to show the potential of the additional multimodal functionality. Special attention is paid to the usability aspects of such a service.

---

* Authors ordered alphabetically

Multimodal services cannot be deployed unless the underlying networks fulfil the requirements imposed by those services. These requirements include the support for simultaneous transmission and reception of voice and data within the same service session as well as the availability of mechanisms for coordination and synchronization. We found that wireless local area networks (WLAN) which are now commonly available, closely satisfy those requirements. Therefore the service was implemented on such a network.

The feedback of the user interaction will be very important to ensure the appetency of users to adhere and subscribe to multimodal services in the near future [3]. The introduction of such services will cause impact on business cases for mobile services. Telecom operators may need to implement new charging models based on the number of modalities subscribed for the service interaction or the number of transactions per modality. Combinations of different charging models are also open possibilities that need further insight that is also expected to be facilitated by the results of the user evaluation.

This paper includes a brief description of the service functionality, our experiences in defining and implementing the demonstrator, the design of the user interface, the results from the expert evaluation and the impact from this evaluation on the design of the service. The results of the user evaluation will be included in the presentation of the paper.

2. The functionality of the MUST tourist guide

The MUST tourist guide for Paris combines speech and pen at the input side, and text, graphics, and speech at the output side. The basis of the service is the equivalent of a printed tourist guide that provides information about a small section of the city, and that uses a detailed map of that section as a navigation and orientation aid. In addition, we have possibilities for extra functionality since this is an online guide, for example using a web based Question/Answering-system.

The tourist guide is organised in the form of small sections of the town around “Points of Interests” (POI’s), such as the Eiffel tower, the Notre Dame, etc. These POI’s are the major entry point for navigation. When the user selects one of the POI’s a detailed map of the surroundings of that object is displayed on the screen of the iPAQ (Figure 3). Many map sections will contain additional objects that might be of interest to the visitor. By pointing at these objects on the screen they become the topic of the conversation, and the user can ask questions about these objects, for example “What is this building?” or “What are the opening hours?”. The user can also ask general questions about the section of the city that is displayed, such as “What restaurants are there in this neighbourhood?". The latter question will add icons for restaurants to the display, and a single restaurant can be turned into the topic of conversation by pointing and asking questions, such as the type of food that is offered, the price range, opening hours, etc. Simultaneous coordinated interaction allows pointing and speech to overlap in time. The information returned by the system is rendered in the form of text, graphics (maps, and pictures of hotels and restaurants), and text-to-speech synthesis.

The service can handle out of database requests. If the system do not find a proper answer to a question about a POI, a multilingual Question/Answering (Q&A) system [2], developed by France Télécom R&D, tries to fetch the answers from the Internet. The access to the Q&A system allows a graceful failure providing a solution in case of out-of-database questions (although it is evident that there also remain unresolved issues in the fields of automatic speech recognition and natural language understanding to assure the correct handling of any out-of-database query).

3. Architecture of the MUST platform

MUST set out to investigate implementation issues related to coordinated simultaneous multimodal input, where *all* parallel inputs must be combined in order to interpret the user's input. In our implementation we opted for the so-called “late fusion” approach, where pen and speech recogniser outputs are combined at a semantic interpretation level. The temporal relationship between different input channels is obtained by considering all the inputs arriving the system within a pre-defined time window. The duration of this time window is a variable that maybe adjusted according to the sequence of inputs (i.e. pen click followed by spoken utterance or spoken utterance followed by pen click) and the current dialog state.

The overall architecture of the MUST-demonstrator consists of a relatively complex application server and a thin client (see Figure 1). The application server is based on a modular architecture comprising six

independent modules that communicate with each other through the GALAXY Communicator Software Infrastructure.

A typical signal flow through the system is as follows: The spoken utterances are forwarded to the speech recogniser module (ASR) by the telephony module (PHN). The pen inputs are transferred from the GUI-client via the WLAN-connection to the GUI-Server. The inputs from the speech recogniser and the GUI-server are grouped in the Multimodal Server (late fusion) and passed to the Dialogue/Context Manager (DM). The DM interprets the result and acts accordingly, for example by contacting the Map Server and fetching the information to be presented to the user. The information is further sent to the GUI Server and Voice Server via the Multimodal Server that performs the fission, i.e. breaking up the output according to the modalities (presentation format – voice and/or graphical output) that is suitable for the user.

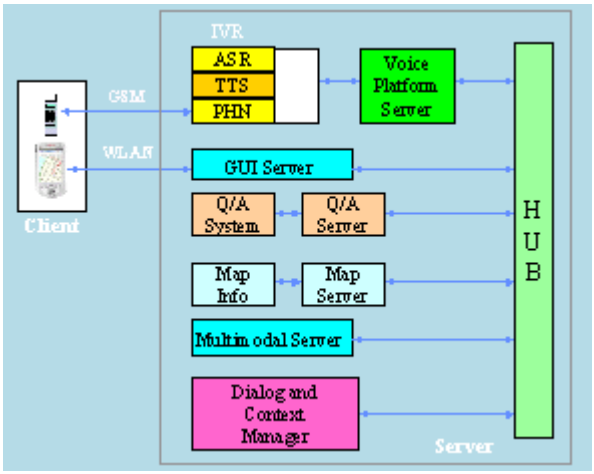

Figure 1 - Overall architecture of the MUST tourist guide application

The following sections provide more details of the modules in the application server and the GALAXY HUB.

3.1. The communication HUB

The communication HUB is provided by the GALAXY communicator which is a public domain reference version of DARPA Communicator maintained by MITRE (http://fofoca.mitre.org). The main features of the GALAXY framework are modularity, distributed nature, seamless integration of modules and flexibility in terms of inter-module data exchange (synchronous and asynchronous communication through HUB and directly between modules).

The GALAXY platform allows us to 'glue' the modules in the application server together in different ways by providing extensive facilities for passing messages between the modules through a central 'Hub'. A module can very easily invoke a functionality that is being provided by another module without knowing which module provides it or where it is running.

The HUB is essentially a facilitator commonly found in agent-based environments. Its operation can be controlled by a script or allowed to run completely autonomously. In MUST the HUB control is script based. In order to keep the format of the messages simple and flexible it has been decided to use an XML based mark-up language named MxML - *MUST XML Mark up Language*. The MxML is used to represent most of the multimodal content that is exchanged between the modules.

Other required parameters for operations such as setup, synchronization, and disconnection are transferred using plain Galaxy messages which are based on a key pair (name plus value) type of data structure.

3.2. The Voice Server

The Voice Server module handles and processes speech. We have developed two different versions of the Voice Server: . One based on the "InoVox" IVR platform [1] developed by Portugal Telecom Inovação, and another version based on the "Tabulib" voice platform [6] developed by Telenor R&D.

These platforms support interfaces to plain telephony (ISDN and/or analogue line), and advanced voice resources such as Automatic Speech Recognition (ASR), and Text-to-Speech Synthesis (TTS). The ASR applied in both voice servers is the SpeechPearl2000 from Philips that supports all the languages used in the project (English, French, Portuguese and Norwegian). ASR-features such as confidence scores and N-best lists are also supported. The TTS-engines however are different for the different languages. Portugal Telecom Inovação's voice platform uses a commercial engine, i.e. the RealSpeak from ScanSoft, while Telenor and France Télécom use proprietary engines developed in their labs.

An important feature of the messages sent or received by the Voice Server is that they are asynchronous. Thus, the module that has sent a message to the Voice Server does not wait for an answer or an acknowledgement, but it proceeds with its next operation. A potential drawback of asynchronous messages is that it may affect the stability and reliability of a system. According to our approach, all the speech processing functions are provided by the Voice Server that comprise several modules such as ASR, TTS and telephonyeven though we could have implemented these modules as separate modules (separate Galaxy servers). However, lumping them together in a single Galaxy server avoids the exchange of large amounts of data (speech streaming) via TCP/IP connections, thereby improving the response time of the system. Moreover, a voice server is a typical component of a conventional (commercial grade) voice only dialog system. Therefore, it is much easier to use this component 'as-is'. To incorporate the existing Voice Servers in the Galaxy based architecture, we only needed to implement a "wrapper" that is placed between the HUB and the existing servers. This wrapper is responsible for processing the Galaxy messages and invoking the appropriate operation (TTS, ASR or PHN).

3.3. The GUI Server

The GUI Server is the gateway between the GUI Client and the application server. The transmission of content back and forth to the GUI Client is wrapped into Galaxy frames and further transmitted to the Dialog Manager Server via the Multimodal Server.

The feedback from the Dialog Manager is an XML body that reflects what to be displayed on the GUI Client. The GUI Server retrieves the content of the XML body, and wraps this into an HTML format to be forwarded to the GUI Client. The HTML file is actually stored on an HTTP (Web) server, and further fetched by the GUI Client, which is nothing but an advanced and customised web browser.

The XML body from the Dialog Manager Server contains the content information to be rendered on the GUI Client, that is, raw information such as text and images to be displayed, and coordinates for items (e.g. point of interests) on a map. The transformation of the XML body to the HTML file is made though an XSLT (http://www.w3.org/TR/xslt). It is the XSL style sheet that really defines the appearance of the GUI, such as the size of text fields, font types, background colours, the width of borders and combo boxes. With the use of style sheets, the appearance of the GUI display can be easily altered in services where the GUI format is dependent of the dialog context, or the user's profile.

3.4. The Multimodal server

The Multimodal server is responsible for the integration of the semantic representations of user's inputs. The temporal relationship between speech and graphical input channels is handled by considering all the input information received within a pre-defined time window. This information is grouped and packed in a single message and passed on to the dialog manager as a first step in the late fusion process. At this stage the message may contain contradictory elements and the interpretation of the combined contents is left to the dialog manager. The duration of the time window is a variable that can be adjusted according to the dialog state and the receiving order of input modalities.

The multimodal server also performs fission. The message from the dialog manager is broken down into two messages. One of the messages contains the speech information, and is sent to the Voice Server. The other message contains the graphical information, and is forwarded to the GUI server.

3.5. The Question & Answering server

In the normal working mode of the MUST tourist guide the system waits for spoken requests of information. When the voice server detects that the user has said something a semantic representation of its oral sentence is dispatched, through the Multimodal Server, to the Dialogue Manager. The message is then parsed and interpreted by the dialogue manager that checks whether the requested information is included in the service database. If the information cannot be found in this database, the dialogue manager redirects the request to the Question Answering (Q&A) server and notifies the user that information was not immediately available, but that it will try to find it nevertheless. The dialogue manager will not be stuck until an answer is received from the Q&A system (host). The user can proceed interacting with the service and he/she will be notified by the Dialogue Manager when the response to the out-of-database question arrives.

The Q&A system searches for the answer in the Internet. It is obviously inappropriate to try and render complete documents on the iPAQ screen, and leave it to the user to detect the answer to the question. Therefore, the Q&A system analyses the documents that it retrieves in detail, to extract a number of answers, each of which is assigned a score for the probability that it is correct. The answers are such that they can be formulated in a short phrase or sentence. If the Q&A system is not able to find an answer, it will respond with the message that it failed to find the requested information.

The Q&A server is physically located in one single site at the premises of France Télécom R&D, due to its complexity. However, the functionality of the Q&A system can be accessed through the Web, since it is implemented as a Web Service. The Web Service approach implies the use of SOAP(Ref?) formatted messages over HTTP for the communication between the server and the applications that access its functionality through the Internet. The implementation of this communication mechanism directly in the Dialogue Manager would result in additional complexity to the module without any advantage in terms of service performance. So it has been decided to create an independent module, named Q&A Proxy Server, to provide and handle the communication mechanism between the main MUST application server and the remote Q&A server. When the QA proxy server receives a message from the dialogue manager with the question issued by the user, it formats the request in SOAP XML encoding and sends to the QA server using the HTTP protocol. The Q&A server runs a listener that accepts the incoming SOAP calls, reads the information from the XML SOAP packets, and maps them to its own processing logic. The Q&A Proxy server parses the response packet in SOAP XML encoding and extracts the answer according to its own internal logic, which is the answer with the highest score. Then the proxy constructs a message with the answer and sends it back to the Dialogue Manager.

3.6. The Dialog & Context Manager Server

The Dialog & Context Manager module consists of four main components, implemented as classes: (1) Context Manager, (2) User model, (3) System response generator, and (4) XML processor.

The *Context Manager* is the heart of the module. It is a finite state machine that contains four main states, START, POI, GOF and FAC.

- START: The dialog is ready to start.
- POI: User has selected a point of interest (POI).
- GOF: User has selected a group of facilities (GOF). such as a set of restaurants or a set of hotels
- FAC: User has selected one particular facility such as a restaurant or a hotel.

The state machine approach with only a few states was sufficient because of the hierarchical nature of the application. The application consists of several POIs, each of which in turn consists of GOFs. Finally, each GOF comprises a set of facilities. When the user generates an event, a state transition can occur. A state transition is defined by the tuple (S_t, I_t), where S_t is the current state and I_t is the current user input. Each state transition has a well defined end state S_{t+1} and an output O_t.

The *User Model* is an array of concepts whose length is set to a pre-defined value. The concept table is filled using the values output by the speech recogniser and the GUI client that lie within a predefined time window. During the filling operation, input ambiguities were solved, in this way completing the late fusion. Once filled, the concept table defines the current input I_t. If the values in the concept table are $I_t(1)$, $I_t(2)$,…….$I_t(n)$, then the N-tuple ($I_t(1)$, $I_t(2)$, $I_t(n)$) is the current input I_t. The number of different inputs can be prohibitively large, even if the length of the concept table (M) and the number of values a given concept can take (K) is moderate. In our case we have reduced the number of inputs by employing a many-to-one mapping from the original input space to a new smaller sized input space.

The *System response generator* is responsible for generating the O_t. It is essentially a mapping from space formed by the tuples (S_t, I_t). It looks at the current state S_t and the input I_t, and generates an output O_t that contains both speech and graphical content. The output can contain pre-stored strings, parameters extracted from the input itself, and data obtained from the back-end Map Database or the Q&A system. Speech output is generated by concatenating components appropriately (i.e. text to be synthesized or pre-recorded audio). Graphical output is generated as an XMLbody.

The *XML processor* performs the XML operations. Since it is difficult to generate a complex XML string through concatenations, we maintain a DOM (Document Object Model) tree that always represents the current graphical output. This is generated from the previous DOM through tree operations such as deletions and insertions. The XML processor is based on XALAN (http://www.apache.org/xalan).

3.7. Client application

The client part of the demonstrator is implemented on a Compaq iPAQ Pocket PC running Microsoft CE, which is connected to the application server via an 802.11b WLAN connection.. The speech part is handled by a mobile phone. The user will not notice this “two part” solution, since the phone will be hidden and the interface will be transparent. Only the headset (microphone and earphones) with Bluetooth connection will be visible for the user.

4. The user interface of the MUST tourist guide

The graphical part of the user interface consists of two types of maps: An overview map showing all POIs, and more detailed maps centred around each POI.

Figure 2 - Overview map with the POIs

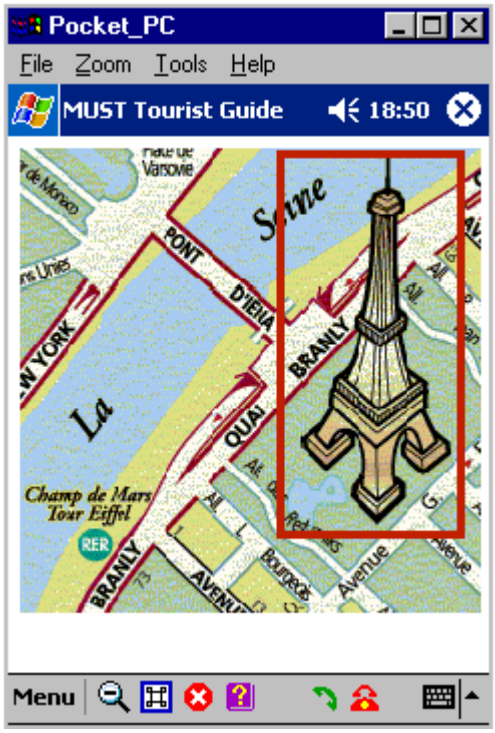

Figure 3 - Detailed map around the POI for the Eiffel Tower.

Pen or speech can be used to display different groups of facilities such as hotels or restaurants. To get information about one specific facility, the user must use pen to select the object, and speech to request the desired information. These two actions can be combined simultaneously, by for example saying "what is the single room rate for this hotel", while tapping on the hotel object. The user can also use speech shortcuts and for example request actions like: " Show me hotels near the Eiffel Tower".

The toolbar shows buttons that are available in the second version of the client application. The first button can be used to zoom out from a detailed map to the overview map. The second button displays a list of selectable facilities. Available facilities are drug stores, hotels, metro stations, restaurants, parking lots and post offices. The third button can be used to cancel/interrupt the action that is currently in progress, for example to interrupt audio output, since "barge in" is not implemented in this version. The third button gives context-sensitive help, while the two remaining buttons are used to connect and disconnect the client with the application server.

5. The expert evaluation

The evaluation of the MUST user interface is carried out in two steps: In phase one a first version of the demonstrator was presented for an expert review. The results from this review were used to design a final version of the interface. Phase two will be a usability test where naive users evaluate the final version.

The expert evaluation had two phases. In phase one, the experts were allowed to explore the interface for a few minutes. The methodology used for the second phase is a Usability Inspection method called Cognitive Walkthrough Method (CWM) [8]. The CWM-technique can be used to evaluate a complete system, a prototype, a system where the UI is not yet complete, or only a specification. The expert evaluation took place at the premises of Telenor R&D and Portugal Telecom Inovação involving Norwegian and Portuguese experts in human-machine interaction. Results should be interpreted with due caution since only twelve experts participated in the evaluation. There were no clear differences between the observations and remarks of the Portuguese and Norwegian experts. The most relevant observations are presented here.

The results from the explorative phase seem to indicate that frequent users of PC and PDA (most of the experts belonged to this group) tend to use a single modality (pen or mouse) to select objects or navigate through maps or menus, even if they are told that it is possible to use speech or both modalities simultaneously. They will have to go through a learning process to get accustomed to the new simultaneous coordinated multimodal interaction. However, after being acquainted with this new interaction mode the learning curve seems to be steep. It was obvious for the experts that the service was multimodal, although the simultaneous use of different modalities was not very intuitive for them at the first sight. This clearly indicates that for the naïve users evaluation we should pay attention to the introduction phase where the service and the interaction mode are explained.

Timing relation between pointing and speech has been subject of study in other experiments [7]. The typical behaviour of the experts in our study was to tap at the end or shortly after the spoken utterance. This was specially the case for utterances ending with deictic expressions like “here” or “there”. The timing for pen click in the absence of this kind of expressions was more spread over the sentence. The timing between pen and speech will be studied further in the end-users evaluation.

The results from the expert evaluation can be organised in two groups. The first considers observations of the expert’s interaction style (which modes they used and the timing between pen and speech). The second group are specific usability issues related to the MUST application, mainly comments on icons, how selected objects were represented in the map, system feedback, prompts, the location of the POIs on screen, etc. These inputs are quite important for tuning the application interface before running the service end-user evaluation. Most experts agreed that without some initial instruction and training it is unlikely that naïve users will start to use a simultaneous multimodal interaction style, although they also agreed that this style may be very efficient They also believe that the users will probably be able to use such interaction style, once they are aware of the system’s features and capabilities. This is also supported by our observations of the experts during the exploration phase. The lack of multimodal applications/interfaces for the general public stresses the need for tutorials to introduce explicitly the simultaneous coordinated interaction mode features before users start using these services. According to the experts a short video or animation will be suitable for this purpose. This is an issue that is going to be studied deeper during the user evaluation scheduled for mid September 2002. The type of introduction that is going to be supplied before users start using the tourist guide will be the main varying parameter in the experiment. Another issue pointed out by the experts is the importance of a well designed help mechanism in a speech-centric user initiative information services. In these systems it is difficult to convey information about its capabilities and limitations [11]. A context dependent help function will be implemented in version 2.

6. Conclusions

We have designed and implemented a multimodal service for small mobile terminals. The advanced simultaneous-coordinated multimodal solution was developed in a relatively short time by using state-of-the-art technologies. However, the service we have developed lies far from a realistic service offered over a 3G mobile network such as UMTS and there are numerous technical challenges to overcome in achieving this goal.

Usability experts in Norway and Portugal have evaluated our multimodal tourist guide. The main findings were:

- Since the simultaneous coordinated multimodal interaction style is entirely new, people need to be told that it actually is possible e.g. to talk while tapping. A short video or animation will be suitable for this purpose.

- When users are aware of the system’s features and capabilities, the users will utilise the simultaneous coordinated interaction mode in a natural way to achieve their information more efficiently than with speech only or pen only.

In the user evaluation (scheduled for mid September 2002) we will study more closely the type of introduction that is needed before users start using the tourist guide.

A significant increase of Multimodal applications is expected in the coming years, bringing benefits to businesses, service developers, telecom operators, and end-users. In the near future, newly developed devices, like PDAs and cellular phones will be available that will be capable to support multiple modes of access and communication, increasing people's mobility. Multimodal applications will be a key component to make this “anyplace, anywhere” access more convenient and real by allowing end-users to select the most appropriate and suitable input and output modalities for interacting according to usage-context. This freedom in terms of modality selection gives rise to new challenges for service and interaction design. Since users are not yet used to multimodal and speech interaction, ways should be invented for the seamless integration of these interaction modes into services.

Probably the best use of Multimodal applications will be in the next generation of wireless networks. 3G wireless networks and beyond will allow greater bandwidth, always-on connections and simultaneous voice and data channels with reduced latency. Delay in the underlying networks is a critical factor for service quality especially for delay sensitive services like voice and video. The networks should maintain good performance even in massive service usage conditions and heavy and "bursty" data traffic levels.

The ultimate aim of multimodal applications will be to create less error-prone, easy to use and natural interfaces to end-users. The current state-of-art puts this goal a little bit far but the technology is evolving very quickly. The future of human-machine interaction will bring more natural interfaces that will become so common as today's mouse, keyboards and display monitors.

7. References

[1] Azevedo,J,, Beires N. (2001) InoVox – MultiService Platform Datasheet, Portugal Telecom Inovação, 2001.

[2] Boualem, M. and Filoche, P. (n.y.) Question-Answering System in Natural Language on Internet and Intranets, *YET2 marketplace*, http://www.yet2.com/

[3] Boves, L., and Den Os, E. (2001) Usability of a Speech Centric Multimodal Directory Assistance Service, Proceedings of the International Workshop on Information Presentation and Natural Multimodal Interaction, Verona, Italy.

[4] EURESCOM P1104 MUST Deliverable 1 ”Multimodal Services – a MUST for UMTS”(2002), http://www.eurescom.de/~pub/deliverables/documents/P1100-series/P1104/, January 2002.

[5] EURESCOM (2002) *Multimodal and Multilingual Services for Small Mobile Terminals.* Heidelberg, EURESCOM Brochure Series.

[6] Knudsen, J.E., Johansen, F.T. and Rugelbak, J.(2000) Tabulib Reference Manual, Telenor R&D scientific document document N-36/2000, 2000.

[7] Martin, J.-C.Julia,L. and Cheyer, A. (1998) A theoretical framework for multimodal user studies, CMC.’98, pp. 104-110, 1998.

[8] Nielsen, J. and Mack, R.L. (eds) (1994) Usability Inspection Methods, Jon Wiley & Sons, Inc, 1994.

[9] Walker, M.A., and Passonmeau, R. (2001)“DATE: A Dialogue Act Tagging Scheme for Evaluation of Spoken Dialogue Systems.Human Language Technology Conference, San Diego, March 2001.

[10] Washster, W. Reithinger,N.,Blocker, A.(2001) Smartkom:Multimodal Communication with a Life-Like Character EuroSpeech, Aalborg, Denmark, pp 1547-1550,September 2001.

[11] Walker, M. A., and Passonneau, R. (2001) DATE: A Dialog Act Tagging Scheme for Evaluation of Spoken Dialog Systems. *Human Language Technology Conference*. San Diego, March 2001

[12] W3C, Multimodal requirements for voice markup languages, W3C working draft July 2000.