D8.6 State of the Art

Contract no	AAL-2009-2-068
Start date of project	24/02/2012
Due date of deliverable	T30
Completion date of deliverable	June 2013
Lead partner for deliverable	Microsoft
Type of version	v1.0

NATURE OF THE DELIVERABLE		
R	Report	**X**
P	Prototype	
D	Demonstrator	

Project co-funded by the European Commission within the AAL Program, call 2		
Dissemination Level		
PU	Public	
PP	Restricted to other program participants (including AALA)	X
RE	Restricted to a group specified by the consortium (including AALA)	
CO	Confidential, only for members of the consortium (including AALA)	

Document History

Issue Date	Version	Change Made / Reason for this Issue
17/06/2013	V1.5	First version
17/06/2013	V1.6	Second version – Reviewed by INESC-ID

Document Main Author	Jairo Avelar (MSFT), António Teixeira (UAVR)
Document signed off by	Daniel Gonçalves (INESC-ID)

Table of Contents

Purpose

The purpose of the current document is to collect the existing state of the art on technologies that are relevant in the scope of the PaeLife project, namely, multimodal user interface, speech technologies and assistive technologies for the elderly.

This document intends to give valuable input to the development of the project, which will allow the creation of innovative means of interaction, and ultimately, the development of the Personal Life Assistant and underlying technologies. This will contribute to the evolution of the current state of the art in assistive technologies for the elderly that resort to multimodal user interfaces.

Overview

In this document we start by analyzing the state of the art in multimodal user interfaces, focusing on the theory behind these, and its main concepts, such as input modalities, modality fusion and fission, dialogue managers and output modalities.

The second section of this document gives further input on the state of the art in elderly speech technologies, namely speech recognition and silent speech interfaces.

Finally, the last section discusses several advancements in the state of the art of assistive elderly technologies, namely speech-enabled and non-speech enabled accessibility applications, commercial solutions already available in the market, products that allow for gesture interaction and actual accessibility hardware available in the market or as prototypes.

1 State of the Art on Multimodal User Interfaces

Introduction

Multimodal access is a proposed new human-machine interface which aims to improve the accessibility of content. This new concept allows an integrated use of various forms of interaction (e.g., sound, gesture, GUI, etc.), simultaneously. These types of interaction also intend to create an environment where a user accesses, transparently, to the same content, regardless of the device (e.g., mobile phone, PDA, computer, etc.).

"A Multimodal User Interface (MMUI) allows a user to interact with a computer by using his or her own natural communication modalities, such as speech, pen, touch, gesture and eye gaze, etc., just as in human-to-human communication. Due to the mutual disambiguation inherent to an MMUI, it has the potential to function in a more robust and stable manner than unimodal systems, which only support single recognition-based technology" [1].

Multimodal interaction **constitutes a key technology** for intelligent user interfaces (IUI). The possibility to control devices and applications in a natural way enables an easier access to complex functionality as well as infotainment content. This kind of interaction is particularly suited for use in automotive scenarios where additional restrictions with respect to input and output have to be taken into account [2].

Multimodal interfaces - which can be considered initiated by the classic work of [3] - were the subject of considerable interest and attention in recent years, leading to the creation of various frameworks. Nonetheless, while offering new possibilities, they also bring new problems and as such, lead to the necessity of a paradigm shift. Several myths [4] still need to be overcomed by creating more knowledge during the use of these technologies, development of theories and development tools. The rise of mobile computing devices has created the need for ubiquitous Web access. This would create the possibility of interaction through various methods which would be beneficial in many usage scenarios [5].

Advantages of Multimodal HCI

Studies, such as [6, 7], have shown that multimodal solutions are typically superior to traditional GUI based solutions or unimodal based interfaces, especially in navigational tasks.

Another advantage of MMUIs is the ability for a user to choose the better suited modality to perform the task at hand, thus improving stability and the robustness of recognition based systems, in situations where a certain modality might have a high error rate (e.g., the use of voice commands in a noisy environment). The possibility to alternate between individual

input modalities is also another advantage of these systems, making it possible to avoid injuries caused by overuse of a single modality during long periods of computer use [8, 9].

One of the most pervasive applications of MMUIs, and considered by some as the main advantage, is in the accessibility and inclusion area where, some studies [10, 11] have shown that multimodal interfaces improve the usage experience by disabled, elderly or not so technologically-savvy users, providing the user with a way to choose among the available modalities, according to their specific constraints, thus including users of "different ages, skill levels, native language status, cognitive styles, sensory impairments, and other temporary or permanent handicaps or illnesses" [8, 9].

Recent evolutions

Recent research results in the area of multimodal interfaces have been focused on the development of natural, adaptive and intelligent interfaces which aim to create conditions for machines to communicate with humans in ways much closer to those used by humans to communicate with each other. These interfaces include features such as the ability to respond to speech and language, vision, touch and other senses.

The European Commission (EC) greatly increased research in this field in the first and fifth calls of the 5th Framework Programme for R&D of the European Commission which included strategic objectives in Multimodal Interfaces. The three related projects, CHIL (http://www.chil.server.de) HUMAINE (http://www.emotion-research.net) and SIMILAR (http://www.similar.cc/) are among the most important projects co-financed by EC in the area of multimodal interfaces.

The Swiss National Center of Competence in Research (NCCR) on Interactive Multimodal Information Management (IM2) is one of the 20 Swiss National Centers of Competence in Research (NCCR). IM2 aims at developing natural multimodal interfaces for human-computer interaction and to foster collaboration, focusing on new multimodal technologies to support human interaction, in the context of smart meeting rooms and remote meeting assistants. Archivus, developed in the framework of IM2, is a good example of a research project handling multimodality both at the content and interaction levels. HephaisTK, developed both in the framework of the NCCR IM2 and of the MeModules project is another example which handles multimodality at the interaction level and aims at providing a tool that allows developers to easily prototype multimodal interfaces.

The MMI program, also in Switzerland, comprehends a number of projects such as the IM-HOST project which targets voice-enabled man-machine interaction in noisy environment and the MeModules project which has the objective of developing, experimenting and evaluating the concept of tangible shortcuts to multimedia digital information.

In the US, the CMU Communicator Project, a project by the Defense Advance Research Projects Agency (DARPA), had the objective of promoting Spoken Dialogue System (SDS) development in research institutes. Besides CMU, other participants involved were AT&T, BBN Technologies, IBM, MIT, University of Colorado amongst others.

Many projects had its origins in the CMU Communicator Project, such as Olympus - a dialog system infrastructure; or Ravenclaw - a two-tier dialog management framework; which will be later detailed in this report.

W3C made a great effort in the area of multimodal interaction for the Web. Instead of implementing a multimodal framework, they propose a set of standard properties - specifically the Extensible Multimodal Annotation Markup Language (EMMA) - that architectures must adhere to.

The W3C Multimodal Interaction Framework is developed by the Multimodal Interaction Activity group within the W3C. This framework aims to create specifications (standards) that allow the construction of tools that can provide access to content, in a multimodal environment, using the Web. This framework is not a development architecture. Rather, it represents a level of abstraction above the architecture. An architecture defines how devices communicate with each other and how they represent information. The Multimodal Interaction Framework collects and organizes the definition of markup languages to specify the information required by the devices/components and to specify the form of information sharing, also between devices/components.

Some of the applications that already exist are based on W3C specifications. Others are built on proprietary formats, which complicates the interaction between competing devices. According to the W3C, the specification of standards for creating multimodal interfaces will produce scalable (and open) standards, thus allowing the rapid development of multimodal interfaces as the technical capabilities of the devices evolve.

Concepts & Some Theory

In this section some base concepts and theory considered important to understand the following sections is, briefly, presented.

Human-Computer Interaction and Media

“Human-computer ‘interaction’ is, in fact, exchange of information with computer systems, and is of many different kinds ...” [12].

This exchange of information is a physical process. We never exchange information in the abstract. When humans exchange information, the information is physically instantiated in some way, such as in sound waves [12].

In fact, humans are traditionally said to have five or six senses for physically capturing information, i.e., sight, hearing, touch, smell, taste, and, if this one is counted as well, proprioception (as when you sense that you are being turned upside down). These are the sensory modalities of psychology [12].

To be perceptibly communicated to humans, the information must be instantiated in one or more of the following six physical media [12]:

Table 1 - Human perceptible information media

Physical Information Carrier	Supported Perceptual Sense	Information Presentation medium
Light	Vision	Graphics
Sound waves	Hearing	Acoustics
Mechanical touch sensor contact	Touch	Haptics
Molecular smell sensor contact	Smell	Olfaction
Molecular taste sensor contact	Taste	Gustation
Proprioceptor stimulation		

Note the non-standard use of the term 'graphics' in English, including not only graphical images, but also ordinary text.

Graphics, acoustics and haptics are currently the all-dominant media used for exchanging information with interactive computer systems [12].

Modalities

Bernsen [12] defines a 'modality' in a straightforward way: a modality or, more explicitly, a modality of information representation, is a way of representing information in some physical medium. Thus, a modality is defined by its physical medium and its particular "way" of representation.

Resulting from the previous definition, we can ask about the physical properties of that medium which make it possible to generate different modalities in it. These properties are called information channels [12]. In the graphics medium, for instance, basic information channels include shape, size, position, spatial order, color, texture, and time.

The notion of an 'information channel' marks the most fine-grained level of Bernsen's Modality Theory (see [12]) and the level at which the theory links with signal processing in potentially interesting ways.

What is a Multimodal System?

Bernsen, in[12], defines a multimodal interactive system as:

> "A multimodal interactive system is a system which uses at least two different modalities for input and/or output. Thus, [IM1,OM2], [IM1, IM2, OM1] and [IM1, OM1, OM2] are some minimal examples of multimodal systems, I meaning input, O output, and Mn meaning a specific modality n".

Correspondingly, an unimodal interactive system is a system which uses the same single modality for input and output, i.e., [IMn, OMn] [12].

An over-the-phone spoken dialogue system is an example of a unimodal system: you speak to it, it talks back to you, and that's it [12].

Other examples are a Braille text input/output dialogue or chat system for the blind, or a system in which an embodied agent moves as a function of the user's movements. There are lots more, of course, if we make creative use of all the modalities at our disposal. Still, the class of potential multimodal systems is exponentially larger than the class of potential unimodal systems [12].

This is why we have to reckon with a quasi-unlimited number of new modality combinations compared to the GUI age.

GUIs are Multimodal - it is probably obvious by now why GUI systems are multimodal: standard GUI interfaces take haptic input and present graphics output. Moreover, both the haptic input and the graphics output involves a range of individually different modalities [12].

The CASE Properties

According to the classification by Nigay and Coutaz, multimodal interfaces can handle inputs in different ways in order to make sense of a set of information provided by the various modalities. The columns of Table 1 represent how modalities may be used by the users of the multimodal interface, while the lines represent the fact that information provided by several modalities may be combined or may be kept independent [13].

		Use of modalities	
		Sequential	**Parallel**
Fu-sion	Combined	Alternative	Synergistic
	Inde-pendent	Exclusive	Concurrent

Figure 1 - Types of multimodal interfaces: two dimensions from the classification space ([13])

Fusion covers the possible combination of different types of data. The absence of fusion is called "Independent" whereas the presence is referred to as "Combined" [13].

The CASE model introduces four properties: Concurrent, Alternate, Synergistic and Exclusive. Each of those four properties describes a different way to combine modalities at the integration engine level, depending on two factors: combined or independent fusion of modalities and sequential or parallel use of modalities on the other hand.

Use of modalities expresses the temporal availability of multiple modalities. This dimension primarily covers the absence or presence of parallelism at the user interface. The granularity for concurrency ranges from the physical actions at the I/0 device level to the task-command level. Absence of parallelism is referred to as "Sequential use" whereas presence is called "Parallel use".

A system that supports "Parallel use" allows the user to employ multiple modalities simultaneously. Conversely, a system characterized by the sequential use of modalities, forces the user to use the modalities one after another.

The CARE Properties

The material for this section comes from [14].

The CARE properties were proposed as a simple way of characterizing and assessing aspects of multimodal interaction: the Complementarity, Assignment, Redundancy, and Equivalence that may occur between the interaction modalities available in a multimodal user interface [14].

The CARE properties are defined as a set of properties that characterize four types of relationships between modalities for reaching states from states.

The formal expressions of the CARE properties rely on the notions of state, goal, modality and temporal relationships. A state (s) is a set of properties that can be measured at a particular time. A goal (g) is a state that an agent intends to reach. An agent is an entity (user, system

or component) capable of performing actions. A modality (m) is an interaction method that an agent can use to reach a goal. A sequence of successive steps is called an interaction trajectory. Two examples of a modality can be the general terms “using speech” or “using microphone”. A temporal relationship (TR) characterizes the use over time of a set of modalities. The use of these modalities may occur simultaneously or in sequence within a temporal window (TW), that is, a time interval.

Equivalence expresses the availability of choice between multiple modalities but does not impose any form of temporal constraint on them. More formally, modalities of set M are equivalent for reaching s' from s, if it is necessary and sufficient to use any one of the modalities.

In contrast to **equivalence**, assignment expresses the absence of choice. More formally, modality m is assigned in state s to reach s', if no other modality can be used to reach s' from s.

Two modalities are used **redundantly** to reach state s' from state s, if they have the same expressive power (they are equivalent) and if they are used within the same temporal window. In other words, the two modalities are required to reach state s’ if they are used redundantly, and they convey the same meaning.

Modalities of a set M are used in a **complementar**y way to reach state s' from state s within a temporal window, if all of them must be used to reach s' from s, i.e., none of them taken individually can cover the target state.

As opposed to Equivalence and Assignment, Redundancy and Complementarity imply fusion of input modalities. The formal expressions of the CARE properties include the notion of temporal relationship (TR) that we further refine by the second dimension of our combination space.

Universal Access to Information Systems

The notion of universal accessibility demands the adaptation of information technology to the user. Above all, disabled persons in a public environment depend on the accessibility to information technology (e.g. cash dispensers, ticket selling machines, etc.). Due to the technological development and the successive intrusion of information technologies into everyday life, “the range of the population which may gradually be confronted with accessibility problems extends beyond the population of disabled and elderly users” [15].

Being accessible requires that a system is able to adapt to the users’ needs, to the task scope and context, and to the technical platform used. An accessible system therefore is a system that is able to optimize its usability depending on the current user, task and system configuration. Universal Accessibility implies that support for users with special needs are not regarded as orthogonal to the application but rather part of the system itself. Users with

disabilities are not considered as a distinct class of users, but rather as part of the continuum of human diversity.

As has been stated by [4] "multimodal interfaces have the potential to accommodate a broader range of users than the traditional interfaces". Providing users with the means of multimodal interaction in their everyday life will enhance accessibility and usability of such systems. Thus, multimodality plays an important role for the integration and rehabilitation of disabled persons also as for the improvement of accessibility of information systems for tomorrow's aging, multilingual and multicultural societies [16].

Smart Homes / Ambient Intelligence

In the future we will be surrounded by smart intuitively operated devices that help us to organize, structure, and master our everyday life. The term Ambient Intelligence, coined by the European Commission's Information Technologies Advisory Group (ISTAG) and Philips, describes this vision. Especially, it characterizes a new paradigm for the interaction between a person and his everyday environment:

Ambient Intelligence (AmI) enables this environment to become aware of the human that interacts with it, his goals and needs. So it is possible to assist the human proactively in performing his activities and reaching his goals [16].

Generic Architecture of Multimodal Systems and Key Components

Considered the first true multimodal interaction system, Richard Bolt's *Put-That-There* work [3], combined speech as a main modality, which allowed a user to specify system commands, with three-dimensional gestures, allowing a user to specify where the action should be applied. This work was not only the first of its kind, but also specified system architecture and concepts that are still used today as the cornerstone of multimodal interaction systems. A graphical representation of this architecture can be seen in **Error! Reference source not found.** .

A multimodal interaction system is composed by input and output devices, their respective recognizers and a group of integration subcomponents, called an integration committee.

The input recognizers are responsible for perceiving the input, and outputting an associated meaning, similar to a semantic processor.

According to [17], "the generic components for handling of multimodal integration are: a fusion engine, a fission module, a dialog manager and a context manager, which all together form what is called the "integration committee".

Error! Reference source not found. also illustrates the processing flow between these components, the input and output modalities, as well as the potential client applications.

As illustrated in the figure, input modalities are first perceived though various recognizers, which output their results to the fusion engine, in charge of giving a common interpretation of the inputs.
When the fusion engine comes to an interpretation, it communicates it to the dialog manager. The dialog manager is in charge of identifying the dialog state, the transition to perform, the action to communicate to a given application, and/or the message to return through the fission component [17].
Finally, the fission engine is responsible for returning a message to the user through the most adequate output modality or combination of modalities, depending on the user profile and context of use [17]
For this reason, the context manager, in charge of tracking the location, context and user profile, closely communicates any changes in the environment to the three other components, so that they can adapt their interpretations [17].

Input Modalities

Inputs of a multimodal dialogue system are a subset of the various modalities such as: speech,

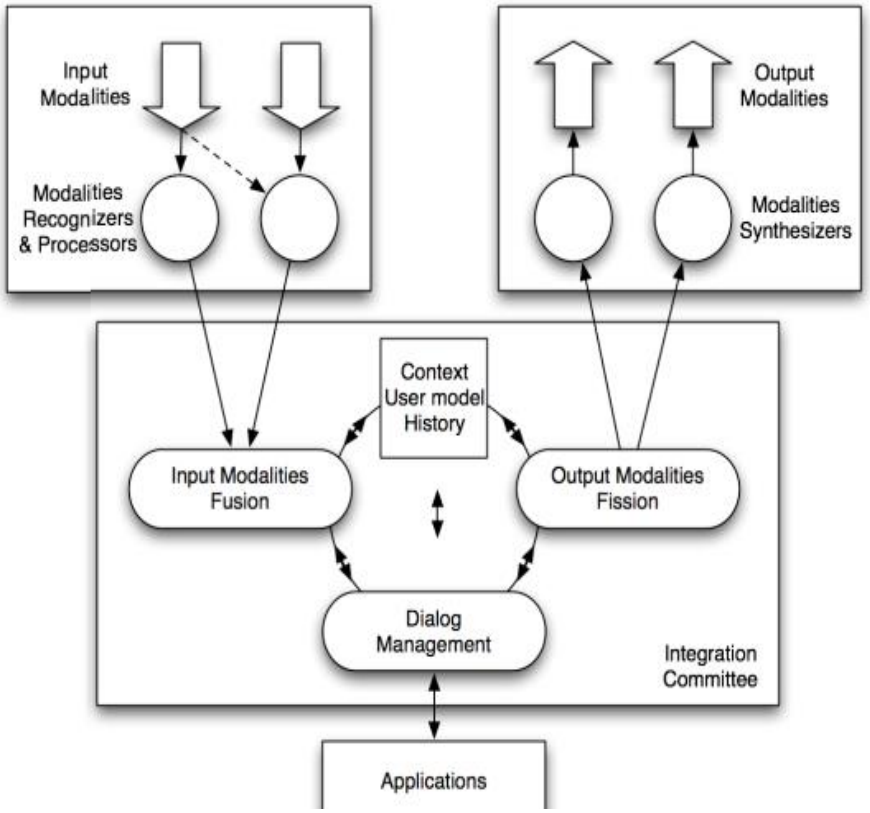

Figure 2 - Architecture of a Multimodal System (DUMAS et al., 2009a)

pen, facial expressions, gestures, gazes, and so on. Two types of input modes are distinguished: active input modes and passive input modes [18].

Active input modes are the modes that are deployed by the user intentionally as an explicit command to the computer such as speech [18].

Passive input modes refer to naturally occurring user behavior or actions that are recognized by a computer (e.g., facial expressions, manual gestures). They involve user input that is unobtrusively and passively monitored, without requiring any explicit command to a computer [18].

Essentially, the Input Modalities (input for Machines, not Humans) are light, sound waves and sensor contacts [12], needing capabilities analogue to Humans vision, hearing and touch (including all levels of processing, as those at the Central Nervous System, CNS).

In this section we review what is presently available to be used as input in multimodal interfaces and the most advanced ways of using. State-of-the-art input modalities are considered outside the scope of this report.

Popular input modalities

A popular set of input modalities are: (1) speech and lips movement, (2) speech and gesture (including pen gesture, pointing gesture, human gesture), (3) speech, gesture, and facial expressions [18].

Input modalities and devices

The smart phone: A ubiquitous input device

The emerging capabilities of smart phones are fueling a rise in the use of mobile phones as input devices to the resources available in the environment such as situated displays, vending machines, and home appliances [19].

The ubiquity of mobile phones gives them great potential to be the default physical interface for ubiquitous computing applications [19].

This would provide the foundation for new interaction paradigms. However, before this potential is realized, we must find interaction techniques that are intuitive, efficient, and enjoyable for applications in the ubiquitous computing domain [19].

Ballagas et al. [19] survey the different interaction techniques that use mobile phones as input devices to ubiquitous computing environments.

Speech (recognition)

Speech is one of the commonly present modality in multimodal systems, appearing as part the 3 most popular combinations mentioned by [18] (already presented in section 0).

Spoken natural dialogue – including speech input - is a key factor for a user-friendly and consistent interaction in intelligent environments [20].

Many recent applications using multimodal interaction – see section 0 on Applications, later in this chapter – explore the use of speech.

On the use of speech as input modality

The advantages of speech are obvious [21]: It is natural, people communicate as they normally do; It is fast (commonly 150–250 word per minute); It requires no visual attention; It does not require the use of hands. Detailed analysis of these claims are presented in [22].

Speech is very resilient as a side channel, making it the ideal mode for "secondary task interfaces". These are interfaces for functions when the computational activity is not the primary task (ex: while driving) [21].

Furthermore, people express themselves more naturally, are less formal and more personal when speaking as compared to writing. It has long been proved that voice is a richer media than written text. It opens up an additional cognitive dimension by introduction of emotion in voice.

About deployment [21], all mobile phones, most of desktop PC and many personal digital assistants (PDAs) are equipped with microphones.

Another aspect in which voice interfaces are advantageous over graphical interfaces, is in mobile environment. Some can't point and click or type with high dexterity. This is especially true for elderly users who had no computer practice when teenagers [21]. With today's proliferation of Web content and mobile phones with broadband data access, spoken interaction could overcome the input limitations of mobile devices [23].

Computer and Mobile applications and many other man-made devices are most often developed for customers without disabilities, not taking into account the requirements of users with special needs, creating new barriers. Speech and Language Technologies can contribute towards the removal of this barrier, but can also do a great deal more by offering individuals with disabilities new means to increase their independence and by encouraging their participation in social and working life. In particular, speech input can create the conditions for greatly improved accessibility [24, 25].

All added, speech presents itself as an ideal solution for human machine interaction especially for elderly people [21].

Pitfals:

Speech input is not always perfect in all situations. Speech is public, potentially disruptive to people nearby, and potentially compromising of confidentiality [21].

The cognitive load imposed by speaking must not be ignored. Generally, when formulating spoken queries, users are not simply transcribing information but are composing it. For such tasks, the real limiting factor may be how quickly one can generate and formulate ideas. This artefact is increased when users are elderly people, because they are introducing more hesitations, increasing the complexity task of translating sound to text [21].

Automatic speech recognition (ASR) engines suffer of lack of performance in real noisy conditions and when the language model which defines the sequence of words is not well aligned and stable. All spoken dialogue system developments will be ruined in its foundations if this fundamental aspect is not considered at this heart of the system [21].

Recommendations

Next Table, from Sun MicroSystems recommendations in this applicative field [21, 26], summarizes the situations fitted and not fitted for speech input.

Table 2 - Voice Input adequacy patterns

Voice input	
Appropriate when . . .	**Inappropriate when . . .**
• there is no other input mode available • no other input mode is practical in the device context (e.g., a key entry or pointing-device system doesn't suit the density of the information that must be input) • no other input mode is practical in the task context (e.g., the task requires the user's hands to be occupied, such as in driving or maintenance and repair) • no other input mode is practical in the user's knowledge context (e.g., users cannot type with sufficient speed, or are illiterate) • no other input mode is practical in the user's physical context (e.g., user's hands or arms are physically disabled)	• the task requires users to talk with others while engaging in it • the environment is very noisy • tasks are easier with other input modes (e.g., choosing from lists)

Engines and Application Programming Interfaces (APIs)

Presently, to have speech as input modality we need: an ASR engine supporting the target language, a way of interacting with the engine (ex: an API), and a mechanism to configure the recognition task (ex: providing a grammar).

Several **speech engines** are available and have been used in Multimodal Systems. They can divided in commercial ones (such as Microsoft, Loquendo, Nuance) and free (such as CMU Sphynx). In general all rely on Hidden Markov Models (HMM) technologies. One of the often used ASR systems in Multimodal research work is Sphynx. To be used in a specific language there is the need for the often called "Language" pack, comprising Acoustic and Language Models. The variety of languages available for each of the engine varies and, at the moment of writing, support for European Portuguese is available from Loquendo, Nuance and Microsoft. For more information on the State-of-the-Art of ASR, see the respective chapter on this document (Chapter 2).

To use the engines functionalities – with emphasis on making words lists available for further processing corresponding to the utterances produced by the humans - toolkit/framework and application developers must make use of an API or other form of communication (such as a client-server communication protocol). The most commonly used APIs are Microsoft Speech API (SAPI), Microsoft Unified Communications Managed API (UCMA) and Java Speech API (JSAPI) [26]. Example of client-server approaches is The Media Resource Control Protocol (MRCP).

JSAPI is a set of abstract classes and interfaces that allow a programmer to interact with the underlying speech engine without having to know the implementation details of the engine itself. Moreover, the JSAPI allows the underlying ASR engine to be easily interchanged with any JSAPI compatible engine. This API was, as an example, used by [27].

The Speech Application Programming Interface or SAPI is an API developed by Microsoft to allow the use of speech recognition and speech synthesis within Windows applications. Several versions of the API have been released, shipped either as part of a Speech SDK, or as part of the Windows OS itself. Applications that use SAPI include Microsoft Office and Microsoft Speech Server. In general the Speech API is a freely-redistributable component which can be shipped with any Windows application that wishes to use speech technology [28].

The Microsoft Unified Communications Managed API 2.0 (UCMA) [29] supports the development of server side, middle-tier applications targeting Microsoft Office Communicator 2007 R2 and Microsoft Office Communications Server 2007 R2. It contains a SIP stack, a media stack as well as powerful speech engines for both automatic speech recognition (ASR) as well as speech synthesis (TTS).

The Media Resource Control Protocol version 2 (MRCPv2) is an IETF recommendation for speech servers communication that relies on Real Time Streaming Protocol (RTSP) and Session Initiation Protocol (SIP) [23]. Loquendo MRCP Server [30] is an example of a server integrated through MRCP. While speech technologies have traditionally been integrated via proprietary APIs, the new method now gaining ground is to rely on MRCP protocol to access a 'speech

server', more suitable for those environments making use of a client-server architecture, such as Automated Call Distribution [30].

VoiceXML can be used for speech to hide the details and providing a higher level API that handles most of the resource management minutiae [23]. However, according to [23], developers are still required to understand the underlying reactive nature of the media resource interaction (ex: prompts in VoiceXML are queued and played only when the execution reaches an input state implicitly defined in the form execution).

Touch

Although touch screens have been available since the 1980's, most of these were used either in a research context or, when commercially available, used in kiosk devices [31]. Nonetheless, the first commercial portable touch screen powered device was only available in the 1990's with the launch of Personal Digital Assistants (PDAs) by Apple (Newton) [32], Palm (Pilot) [33] and later Windows CE and Windows Mobile powered devices [34]. These devices, however, allowed only a single finger or a stylus interaction, commonly called as screen tapping, and also providing the possibility to enter text, via a virtual keyboard on the screen, or via handwrite recognition, using the stylus.

The launch of Apple's iPhone in 2007 marked the main-streaming of a new touch interface concept: multi-touch screens. This technology permits the user to interact with a device using multiple fingers simultaneously, thus allowing more natural gestures on the screen, which are then associated by the device to specific functions. In fact, since touch screens allow users to directly interact by touching the information displayed on the screen, this technology is considered to be one of the most easy to use, even by users with low or no computer literacy [104]. The multi-touch concept has been under development since 1982, with some early commercial multi-touch capable devices dating back to at least 2001 [35]. Currently, many devices exist with multi-touch screens, ranging from the several iterations of Apple's iPhone, and some Android powered smartphones [36], to some tablet PCs running Windows 7, and Microsoft's Surface [37].

"In recent years we have witnessed an exponential growth of technologies to support direct-touch interactive surfaces" [38].

Another advantage of using touch screens is that the input device is also the output device, saving work space by not requiring any accessories (e.g., keyboard or mouse). Being able to touch, feel and manipulate objects on a computer screen, in addition to seeing and hearing them, provides a sense of immersion [104]. Murata et al. [105] argues that, in comparison with a traditional mouse and keyboard setup, the touch panel has the advantage of simplicity and offers opportunities to design more accessible systems.

The touch screen also allows for faster selection of menu items and less issues with peripherals (such as a mouse giving out at a crucial moment). By utilizing the correct touch screen monitor for the application, hospitals can speed information and crucial machine settings to the equipment or personnel it needs to reach, without having to sit down and type it out on a conventional keyboard. Point of sale touch screens speed up checkout for customers and move lines along much smoother than if every selection on the screen is to be tracked and selected by a conventional mouse [39].

The touch screen interface can be beneficial to those that have difficulty using other input devices such as a mouse or keyboard. When used in conjunction with software such as on-screen keyboards, or other assistive technology, they can help make computing resources more available to people that have difficulty using computers [39]. Indeed, since this technology relies more on software than on hardware, it is very versatile and can be easily adapted to particular users' needs. For example, considering the elderly capabilities, this kind of interfaces can have some specific features like multiple sizes for fonts, buttons and icons, as pointed out in [106], thus increasing the accessibility of the system.

However, these new and updated technologies also present some disadvantages. Since they lack the haptic feedback of physical buttons, it can be harder to accurately select targets, especially if they are small. This characteristic hampers certain tasks, such as text-entry, where the user has to constantly select one of many small targets [107].

DiamondTouch, one of the earlier multi-touch systems, allows multiple users to simultaneously interact on a tabletop. Through capacitive coupling it associates touch regions with each user, useful for supporting user-specific operations. The rough shape of each finger contact is determined through an antenna matrix [38].

Computer-vision-based technologies (ex: Frustrated Total Internal Reflection (FTIR), which recovers the surface regions being depressed by fingers) are also widely employed to enable direct-touch surfaces. An alternative approach is based on Diffuse Illumination (DI), which detects not only the contact regions but also fingers hovering above the surface within a certain distance. This is used on systems such as the Microsoft Surface [37, 40]. Compared to capacitive-based sensing, vision-based systems provide higher fidelity in detecting the contact shape [38].

Leveraging additional information available on the surfaces could potentially result in richer and novel interactions. In their work, [38] specifically explore the role of finger orientation. According to them, this property is typically ignored in touch-based interactions partly because of the ambiguity in determining it solely from the contact shape. As such, they present a simple algorithm that unambiguously detects the directed finger orientation vector in real-time from contact information only, by considering the dynamics of the finger landing process. They demonstrate how finger orientation can be leveraged to enable novel

interactions and to infer higher-level information such as hand occlusion or user position. Finally, they present a set of orientation-aware interaction techniques and widgets for direct-touch surfaces.

NextWindow's touch Application Program Interface (API) [41] provides programmers with access to touch data generated by a NextWindow touch screen. It also provides derived touch information. For information about the capabilities of NextWindow products and which ones you can interface to using the API, see NextWindow Latest Technical Information. The touch events, data and derived information, can be used in any way the application wants. Communications are via HID-compliant USB. The API is in the form of a DLL that provides useful functions for application developers.

Considering the elderly and their deteriorated capabilities (degraded vision and tactile sense), this kind of interfaces should have some specific features like multiple sizes for fonts, buttons and icons, as pointed out by Stone [Stone 08]. The author verified that one of the main problems of mobile touch devices among elderly, is that buttons are too small. But since the button size and arrangement is under software control, it is possible to circumvent that problem.

In a study conducted by Werner et al. [108], he selected 11 seniors with no previous internet or PC experience and evaluated the general usability and acceptance of a selected tablet. The results of the study show high acceptance and satisfaction rates among the user group and hence suggest a future focus on the development of tablet based applications for seniors. The authors argue that tablets are an easy way to step into the digital world.

Loureiro et al. [109] analyzed different aspects of 8 touch-based tabletop interfaces for the elderly. In all surveyed works, they concluded that touch and gestures yield a natural, direct, and intuitive way of interaction with a device allowing easier human-computer interaction for elder users. In every surveyed work, they found that the required computer literacy from the users is very low. Indeed, people with low or no computer literacy found using touch screens easy and motivating.

Mouse simulation

To ensure compatibility with traditional legacy applications, researchers have studied cursor control and mouse simulation techniques. The DiamondTouch-mouse supports a right-click by tapping with a second finger. DTMouse further enhances the functionality of the DiamoundTouch-mouse by addressing issues such as mouse-over, smooth toggling of left mouse button, ergonomics and precise input. In DTMouse, states of the mouse were determined based on timeout intervals of holding a finger down [38].

Gestures

Gesture recognition is a **wide field** of different interaction techniques and devices that have the common goal of interpreting human bodily motions into computer input for interaction. It is one of the most natural and intuitive ways of interacting with technology, since it closely mimics how humans interact with each other. The gesture recognition term is often widely used to encompass anything from the historical mouse motion gestures, to the more modern accelerometer based physical gesture recognition.

Three dimensional gesture interfaces are used as a natural way of interacting with computer systems, usually through the interpretation or recognition of human gestures originating from facial expressions or hand gestures. These types of interfaces make it possible to explore the potential of human body language, thus allowing a more expressive way of communicating with computers. Due to the very expressive nature of gestural interfaces and the multitude of gestures that can be elaborated, probability based techniques such as, Hidden Markov Models (HMM) have to be used to better interpret the meaning of these gestures.

According to [42], there are two ways for recording gestures. Non-instrumental projects recognize hand and finger postures with cameras and image processing algorithms. Other projects use instruments for recording, for example sensor gloves or hand devices with integrated sensors like accelerometers or gyroscopes.

Gesture recognition interfaces gained popularity in the video game industry. The first popular gesture and movement recognition device was the Wii Remote for Nintendo's Wii console, a remote shaped accessory released in 2006. A main feature of the Wii Remote is its motion sensing capability, achieved through the use of accelerometer and optical sensor technology, which allows the user to interact with and manipulate items on screen via gesture recognition and pointing. In 2007, Wii Balance Board was introduced, an accessory shaped like a household body scale. The board contains four pressure sensors that are used to measure the user's center of balance and weight.

A similar device is the Playstation Move for Playstation 3, released in 2010. It is based around a handheld motion controller wand, and uses the PlayStation Eye (a digital camera device similar to a webcam) camera to track the wand's position. Inertial sensors in the wand detect its motion.

On the other hand, Microsoft's Kinect for the Xbox 360 relies solely on a color and a depth camera to detect users' movements. These cameras enables users to control and interact with the Xbox 360 without the need to touch a game controller, through a natural user interface using gestures and spoken commands. Kinect can also be used as an input device for Windows PCs. Since Microsoft released the Kinect software development kit for Windows 7 on 2011,

many independent projects emerged[1], which range from presentations control interfaces [110] to interactive dressing rooms[2].

One of the greatest advantages of this kind of interface is that it does not require user proximity to the processing equipment, allowing users to freely move in the captured area. Video processing interfaces are more natural over the ones that require a motion sensing device, since they do not require any other external peripherals other than the video camera. Besides, this type of accessories usually requires the use of buttons to perform certain actions, which makes them less natural to use. Video recording also keeps the users hands free to do other things.

Nevertheless, this kind of interface should be used to interpret gestures that are natural and commonly used in the real world, and not the other way around: users should not have to remember a certain gesture to perform a certain action. Only this way we can preserve the naturalness of this interface.

The tool LiveMotion from AiLive is a framework for Wii game developers focused on learning and recognizing more complex gestures. The creation of motion recognizers is mastered by showing gesture examples without coding or scripting. Recognition should be very fast and without using buttons but is only usable by game developers who have a contract with Nintendo.

Wii Remote Acceleration Sensors, an ADXL330 accelerometer, is integrated in the Wii Remote controller [42]. It measures acceleration values with 3 axis sensing in the interval -/+ 3g. The acceleration is described in a right-handed Cartesian coordinate system. A Wii Remote, which lies bottom side down on a table, measures the value of 1g in the direction of the z--axis. This is the force the hand needs to exert against gravity and thus an unmoved Wii Remote always measures the absolute acceleration value of 1g. In free fall the absolute value is zero. The complete movement of the hand within the three-dimensional space can be described by observing acceleration in a series respective to the time.

Some novel technologies are currently under development such as Microsoft's Project Natal, which allows full-body 3D motion capture as well as facial recognition [43], or MIT Media Lab's BiDi screen, which allows interaction with devices, similar to that idealized in Minority Report [44].

A different kind of gesture recognition interface, based on the detection of human muscle movement in real-time through the use of forearm electromyography (EMG), has been tested

[1] http://www.kinecthacks.com/ and http://www.kinecthacks.net/

[2] http://www.kinecthacks.com/kinect-fitnect-interactive-dressing-room/

with positive results, thus allowing gesture recognition in situations where conventional hand gesture recognition wouldn't be very easy to use [45].

Advantages and Issues

Gesture recognition interfaces offer many advantages when compared with more standard input devices such as keyboards and mice. Among these are ease of use by users with motor impairments, requiring less dexterity, as well as allowing a more interactive and immersive operation of multimedia applications such as games.

Some technical limitations must also be taken into account, such as accuracy of gesture recognition technologies being used. Microsoft's Project Natal for instance, due to having its recognition hardware slightly offset from the display, cannot be properly used at short distances [46]. Issues like hardware sensitivity, image noise, environmental lighting or background items, also make gesture recognition more difficult to accomplish.

Accelerometer based gesture recognition.

As an example, the iPhone platform is fitted with a 3Axis MEMS ±2g/±8g accelerometer. Through the ObjectiveC APIs provided by the iPhone SDK we can access the output of this device in ±2g mode at a rate of up to 200Hz.

The GA Tech Gestures and Accelerometers Recognizer Toolkit (GART), previously named GT2K , detailed in [47] is a Java based toolkit that utilizes HTK and wraps it with a framework that is designed to enable swift development of gesture recognition applications.

GART allows a quick way of implementing an accelerometer based gesture interface that can be fed feature vectors and output classified gestures without having to focus on the hidden Markov details.

GART allows fast prototyping of gesture based interaction applications which we can then use to study the implications and capabilities of such an input system.

The framework TaKG is a toolkit for gesture recognition and serves to simplify the integration of gesture controlled interaction into applications. It implements needed functionalities for signal feature extraction and the recognition algorithms like SVM, NN and DTW.

wiigee [48, 49] is an open-source gesture recognition library for accelerometer-based gestures specifically developed for the Nintendo® Wii™ remote controller. It is implemented in Java™ and, thus, is platform-independent. Using a third-party Bluetooth®-library **wiigee** allows to define and recognize your own, freely trained gestures.

The **wiigee** library:

- allows you to **define (train) your own arbitrary gestures**,
- **recognizes these gestures** with high accuracy,
- offers an **event-driven architecture** with which you will be able to integrate the gesture-input as easy as common mouse-input.

Wiigee utilizes state of the art probability theory-methods to deliver reliable results fast and efficient. It is still under development.

Gaze

Gaze or eye tracking is the process of measuring either the point of gaze ("where we are looking") or the motion of an eye relative to the head. To achieve this, an eye tracker is used, which is a device for measuring eye positions and eye movement. There are a number of methods for measuring eye movement. The most popular variant uses video images from which the eye position is extracted. Other methods use search coils or are based on the electrooculogram [111].

Gaze-tracking interfaces consist on a camera focused on one or both eyes. Most modern eye-trackers use contrast to locate the center of the pupil and use infrared and near-infrared light to create a corneal reflection: the video image is analyzed to identify a large bright circle (pupil) and a brighter dot (corneal reflection) and compute the center of each. The line of gaze is determined by these two points. Depending on initial calibration, the vector between these two features can be used to compute gaze intersection. It is also possible to track users' gaze through appearance-based interfaces. It uses images photographed by a computer's camera, and apply computer vision algorithms to track the eye and its orientation in the images [50].

Gaze tracking setups vary greatly: some are head-mounted (sometimes being considered too intrusive), some require the head to be stable (for example, with a chin rest), and some function remotely and automatically track the head during motion. Gaze is commonly used by individuals with severe motor impairments.

A recent and popular commercial eye-tracking system is Tobii I-Series[3], which is controlled through gaze interaction via a built in eye tracker. The eye control unit enables access to the computer in situations where users are unable to use their hands. By looking at a screen, users control the mouse cursor and can click by blinking, dwelling (staring at the screen for a certain length of time) or using a hardware button. The authors state that the Tobii eye control unit operates with a high level of precision in nearly all light conditions, achieving a great accuracy

[3] http://www.tobii.com/en/assistive-technology/global/products/hardware/tobii-i-series/ready-when-you-are/

rate on most users, regardless of eye color, age or ethnic origin, and even those using glasses or contact lenses. There are other similar approaches provided by different vendors.

Advantages and Issues

One of the main advantages of gaze interfaces is that they can be used in situations when the user has severe impairments that prevent him/her from operating any other types of interfaces.

However, as noted in [50], there are some issues with these types of interfaces that must be taken into consideration. While wearable interfaces are considered more accurate in capturing the eye's movement, their intrusiveness can be uncomfortable to the user. On the other hand, non-wearable systems require personalized calibration for each user, which can take quite some time. IR based systems, however, has not yet been considered completely safe, as the long term effects of exposure to IR are still unknown. Also, issues can arise from the usage of low image resolution in two camera appearance-based systems, as they can reduce the accuracy of this method.

Lips movement

Several works addressed automatic lips movement extraction, which can be used as input for multimodal interaction. A few of them are briefly described in this section.

Chan [51] presents an unsupervised image segmentation method to hierarchically locate the users face and then the lips. Techniques employed include modelling in the hue-saturation color space using Gaussian mixture models and the use of geometric constraints. With the region of interest automatically located, the model extraction problem is then formulated as a regularized model-fitting problem. The use of a generic shape as prior information improves the accuracy of the extracted lip model, which is based on a cubic B-spline representation.

Choraś [52] presents a pattern recognition methods and introductory results of automated human lips recognition system. In their research integrated lip shape descriptors and color features had been used to determine human identity. Their system detects lips easily from face **images** captured especially for lip recognition project (lower face only), but struggles with satisfactory lips detection on other datasets, especially face images from surveillance cameras.

The proposed Lip Recognition process is mainly divided into three parts: (1) Lip Detection; (2) Lips Shape Feature Extraction and (3) Lip Color Recognition.

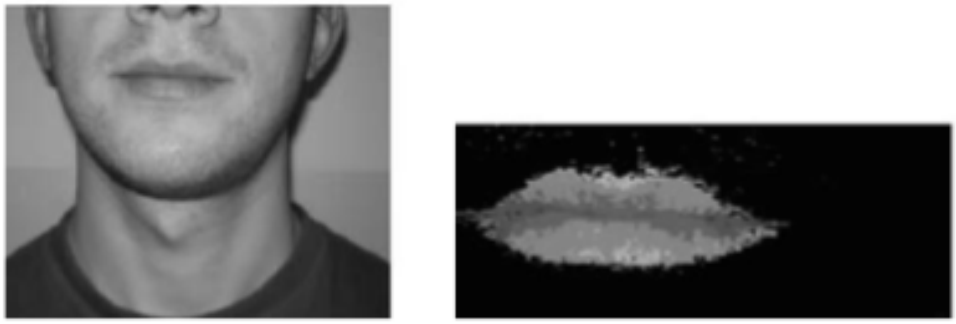

Figure 1- Example of lower face and extracted corresponding lips area, from [52].

In step 1, they first detect the lips form the face images, and then perform segmentation, binarization and size normalization. Figure 1 shows a result example result of this step. In step 2, and after the lips detection stage, shape features of the binarized lips images are calculated. In step 3, they calculate statistical color features in three types of color spaces: RGB, H S V and Y U V. Features are calculated separately for each channel in the used color spaces.

Yoshida et al. [53] have developed a simple **infrared lip movement sensor** (Figure 2) mounted on a headset, and made it possible to acquire lip movement by PDA, mobile phone, and notebook PC. The sensor consists of an infrared LED and an infrared photo transistor, and measures the lip movement by the reflected light from the mouth region. From experiment, it states they achieved 66% successfully word recognition rate only by lip movement features. As such, they claim this experimental result shows that the developed sensor can be utilized as a tool for multi-modal speech processing by combining a microphone mounted on the headset.

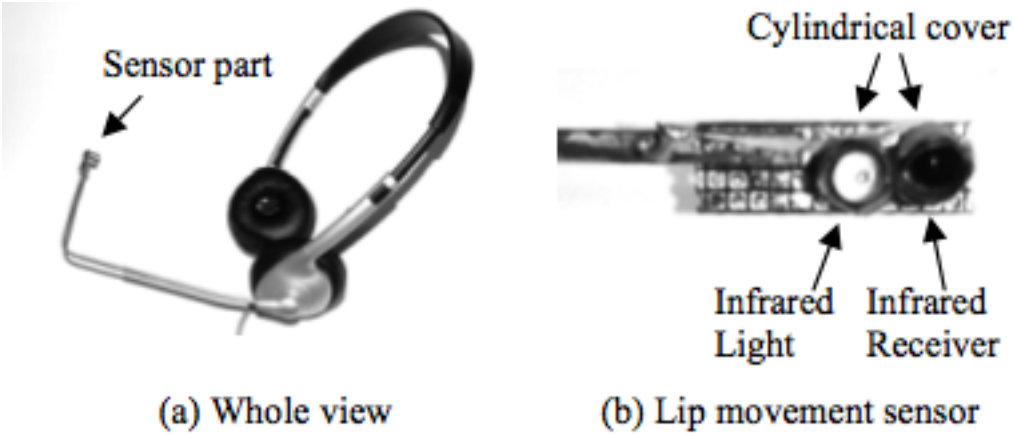

Figure 2 - The infrared Sensor of [53]

Facial expression

According to Cowie et al. [54], facial expression classification approaches could be divided into two main categories: target oriented and gesture oriented. Target oriented approaches attempt to infer the human emotion and classify the facial expression from one single image containing one typical facial expression. Gesture oriented methods utilize temporal information from a sequence of facial expression motion images.

Facial Expression Recognition and Tracking for Intelligent Human-Robot interaction

Intelligent service robots rely on effective utilization of available sensors such as sound and vision sensors to gather information for decision making, planning, and ultimately empathetic interaction with humans. As a crucial component of a social robots sensing suite, a large part of research on social robots has focused on visual data analysis. It involves human/face detection and the fusion of stereo and infrared vision on board social robots with greater flexibility and robustness, for the purposes of attention focusing and for synthesizing more complex social interaction concepts, like comfort zones, into the robots.

In [55], an automated and interactive computer vision system is investigated for human facial expression recognition and tracking based on the facial structure features and movement information. Twenty facial features are adopted since they are more informative and prominent for reducing the ambiguity during classification. An unsupervised learning algorithm, distributed locally linear embedding (DLLE), is introduced to recover the inherent properties of scattered data lying on a manifold embedded in high-dimensional input facial images.

Figure 3 - Results of real-time person independent expression recognition, from [55].

The selected person-dependent facial expression images in a video are classified using the DLLE. In addition, facial expression motion energy is introduced to describe the facial muscles tension during the expressions for person-independent tracking for person-independent recognition. According to the authors, this method takes advantage of the optical flow which tracks the feature points' movement information. Figure 3 shows some results of their work.

Automated Facial Expression Recognition System

Recent advances in facial image processing technology have facilitated the introduction of advanced applications that extend beyond facial recognition techniques. Ryan et al. [56] introduce an Automated Facial Expression Recognition System (AFERS): A near real-time, next generation interrogation tool that has the ability to automate the Facial Action Coding System (FACS) process for the purposes of expression recognition. The AFERS system analyzes and reports on a subject's facial behavior, classifying facial expressions with one of the seven universal expressions of emotion (sadness, disgust, fear, anger, contempt, surprise and happiness).

AFERS employs shape and appearance modelling using constrained local models for facial registration and feature extraction and representation, and support vector machines for expression classification. AFERS provides both pre and post-analysis capabilities and includes features such as video playback, snapshot generation, and case management. In addition to the AFERS processing algorithms, the implementation features a plug-in architecture that is capable of accommodating future algorithmic enhancements as well as additional inputs for behavior analysis.

Figure 4 gives an example of AFERS processing of an image sequence.

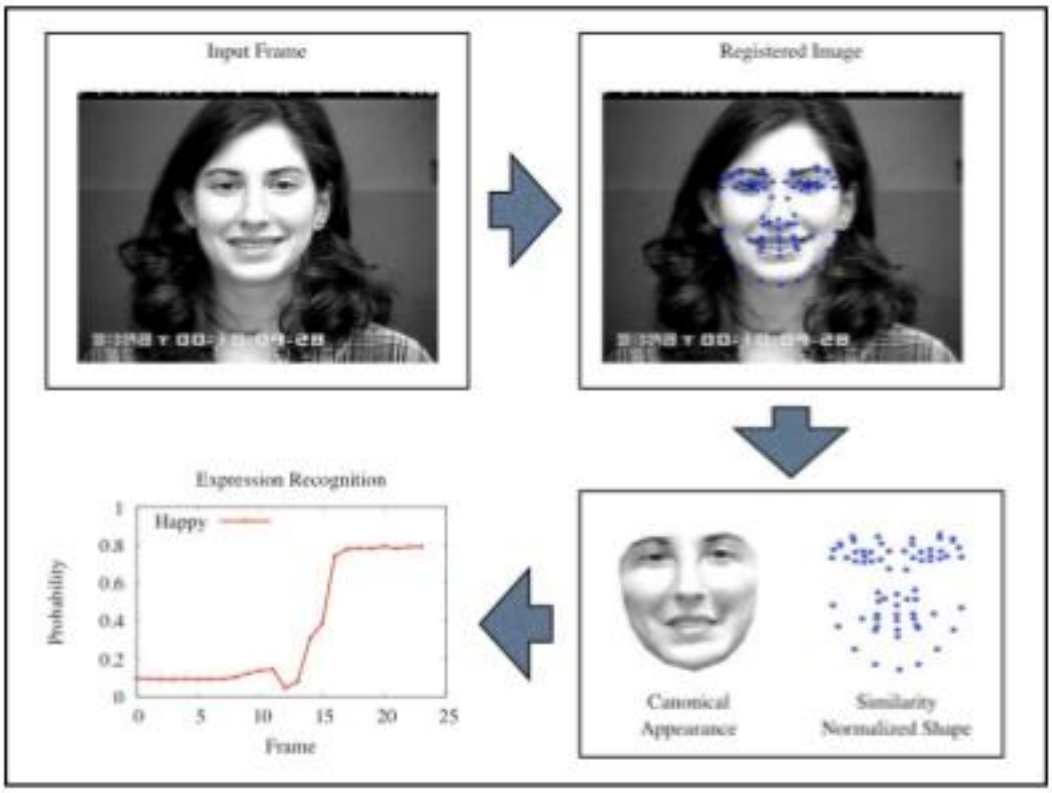

Figure 4 - AFERS processing example, from [56]

Facial Expression Recognition as a Creative Interface

Valenti et al. [57] presents an audiovisual creativity tool that automatically recognizes facial expressions in real time, producing sounds in combination with images. The facial expression recognition component detects and tracks a face and outputs a feature vector of motions of specific locations in the face.

The feature vector is used as input to a Bayesian network which classifies facial expressions into several categories (e.g., angry, disgusted, happy, etc.). The classification results are used

along with the feature vector to generate a combination of sounds and images that change in real time depending on the person' s facial expressions.

Fusion of Input Modalities

Information from various input modalities is extracted, recognized and fused. Fusion processes the information and assigns a semantic representation which is eventually sent to DM [18].

The goal of fusion is to extract meaning from a set of input modalities and pass it to a human-machine dialog manager. Fusion of different modalities is a delicate task, which can be executed at three levels: at data level, at feature level and at decision level. Three different types of architectures can in turn manage decision-level fusion: frames-based architectures, unification-based architectures or hybrid symbolic/statistical fusion architectures.

As further elaborated below, multimodal input fusion can be done in several ways, with varying advantages and disadvantages. Modalities can be processed sequentially or in parallel, and the output of modality processing can be either combined or used independently. Independent modality processing can be useful if applications using this architecture need lower level information regarding the user's input. Modality semantic content combination at the fusion engine level however, allows better abstraction and the addition of other modalities without modifying upper level components in the application's architecture, but there may not be enough information needed by upper levels in the architectural model of the application, and modality disambiguation is more difficult to handle.

Regarding sequential and parallel modality processing, a parallel model is better suited for situations where input data redundancy is important, however, modality synchronization, through timing mechanisms must be taken into account, so as to avoid wrong interpretations of user input. Sequential modality processing allows the development of simpler fusion algorithms, without a critical need for input synchronization. However, it won't be possible to correctly process simultaneous events generated by the user, as semantic interpretation is done individually to each event [13].

Three level fusion execution

Sharma et al., cited by [58], consider these three levels for fusion of incoming data. Each fusion scheme functions at a different level of analysis of the same modality channel.

As an illustration, they consider the speech channel: data from this channel can be processed at the audio signal level, at the phoneme (feature) level, or at the semantic (decision) level.

Data-level fusion is used when dealing with multiple signals coming from a very similar modality source (e.g., two webcams recording the same scene from different viewpoints). With this fusion scheme, no loss of information occurs, as the signal is directly processed. This benefit is also the main shortcoming of data-level fusion. Due to the absence of pre-processing, it is highly susceptible to noise and failure [58].

Feature-level fusion is a common type of fusion when tightly-coupled or time synchronized modalities are to be fused. The standard example is the fusion of speech and lip movements. Feature-level fusion is susceptible to low-level information loss, although it handles noise better [58].

Decision-level fusion is the most common type of fusion in multimodal applications. The main reason is its ability to manage loosely-coupled modalities like, for example, pen and speech interaction. Failure and noise sensitivity is low with decision-level feature, since the data has been preprocessed. On one hand, this means that decision-level fusion has to rely on the quality of previous processing. On the other hand, unification-based decision-level fusion has the major benefit of improving reliability and accuracy of semantic interpretation, by combining partial semantic information coming from each input mode which can yield "mutual disambiguation" [58].

Typical architectures for decision-level fusion are frame-based fusion, unification-based fusion and hybrid symbolic/statistical fusion. Frame-based fusion uses data structures called frames or features for meaning representation of data coming from various sources or modalities. Unification-based fusion is based on recursively merging attribute-value structures to obtain a logical whole meaning representation. Symbolic/statistical fusion is an evolution of standard symbolic unification-based approaches, which adds statistical processing techniques to the fusion techniques. An example of a symbolic-statistical hybrid fusion technique is the Member-Team-Committee (MTC) architecture used in Quickset.

A comparison of the 3 levels is presented in Table 3 - Characteristics of fusion levels, extracted from [58].

Table 3. Characteristics of fusion levels.

	Data-level fusion	Features-level fusion	Decision-level fusion
Input type	Raw data of same type	Closely coupled modalities	Loosely coupled modalities
Level of information	Highest level of information detail	Moderate level of information detail	Mutual disambigua-tion by combining data from modes
Noise/failures sensitivity	Highly susceptible to noise or failures	Less sensitive to noise or failures	Highly resistant to noise or failures
Usage	Not really used for combining modalities	Used for fusion of particular modes	Most widely used type of fusion
Application examples	Fusion of two video streams	speech recognition from voice and lips	Pen/speech interaction

Table 3 - Characteristics of fusion levels, extracted from [58].

Fusion can also be categorized into 'early' and 'late'. 'Early' fusion integrates multimodal inputs after the feature extraction of each input mode. 'Late' fusion follows appropriate interpretation for each mode is determined. It is supposed to integrate related (but complementary) inputs [1]. Early fusion represents fusion at the signal level, and late fusion represents fusion at the semantic level.

"Fusion engines are fundamental components of multimodal interactive systems, to interpret input streams whose meaning can vary according to the context, task, user and time" [13]. A Survey (as 2009) can be found in [13].

Fusion algorithm in PAC-Amodeus

PAC-Amodeus [14] is a conceptual model that has been used to implement the MATIS (Multimodal Airline Travel Information System) system.

The fusion mechanism relies on a uniform representation: a melting-pot which is a 2D structure. On the vertical axis, the "structural parts" model the composition of the task objects. For example, destination and time departure are the structural parts of the task objects handled for MATIS. The horizontal axis represents the time. Fusion is performed on those melting pots. There are three types of implemented fusion in PAC-Amodeus: microtemporal fusion, macrotemporal fusion, and contextual fusion. Microtemporal fusion is used to combine input events produced in parallel or in a pseudo-parallel manner (i.e., Parallelism, Coincidence or Concomitance along the temporal dimension of the combination space). Macrotemporal fusion is used to combine input events produced

sequentially (i.e., Anachronism or Sequence in Figure 2). Contextual fusion is used to combine related input events produced without attention for temporal constraints.

Current implementations

Table 4, from [17] section 2.3, summarizes the major architecture traits of recent implementations of multimodal systems, as well as their fusion mechanisms.

	ICARE [3]	OpenInterface [17]	IMBuilder/MEngine [4]	Flippo et al. [12]	Krahnstoever [16]	Quickset [6]	Callas [1]	HephaisTK [10]
Finite state machine			x					
Components	x	x					x	
Software agents				x		x		x
Fusion by frames					x			x
Symbolic-statistical fusion						x		
CARE properties	x	x						x

Table 4 - Architecture traits of multimodal systems[17]

Dialogue Management & Dialogue Systems

Dialogue Manager is the core module of the system. The main tasks of DM are [18, 59]:

- updating the dialogue context on the basis of interpreted communication
- providing context-dependent expectations for interpretation of observed signals as communicative behavior
- interfacing with task/domain processing (e.g., database, planner, execution module, other back-end system), to co-ordinate dialogue and non-dialogue behavior and reasoning
- deciding what content to express next and when to express it

The term ”dialogue context” can be viewed as the totality of conditions that may influence the understanding and the generation of communicative behavior (Bunt (2000), cited by [18]).

Both spoken dialogue system and multimodal dialogue system need a central management module called the Dialogue Manager [18]. The Dialogue Manager (DM) is the program which coordinates the activity of several subcomponents in a dialogue system and its main goal is to maintain a representation of the current state of the ongoing dialogue [18]. For convenience we divide the presentation of Spoken and Multimodal Dialogue Systems into two subsections, which follow.

Spoken Dialogue Systems

Information presented in this section comes in great part from [60, 61] and the Phd of Marcelo Quinderé on “Comunicação Humano-Robô através de Linguagem Falada” [112]. For more details see [60, 61,112].

Introduction

Over the recent years, advances in automatic speech recognition, as well as language understanding, generation, and speech synthesis have paved the way for the emergence of complex, task-oriented conversational spoken language interfaces. Examples include: Jupiter provides up-to-date weather information over the telephone; CMU Communicator acts as a travel planning agent and can arrange multi-leg itineraries and make hotel and car reservations; TOOT gives spoken access to train schedules; Presenter provides a continuous listening command and control interface to PowerPoint presentations; WITAS provides a spoken language interface to an autonomous robotic helicopter; AdApt provides real-estate information in the Stockholm area; is a spoken-language enabled planning assistant [61].

Current dialog management technologies [61]

A number of different solutions for the dialog management problem have been developed to date in the community. Some of the most widely used techniques are: finite-state, form-filling, information-state-update, and plan-based approaches. The last two are in [60] incorporated in the so-called advanced systems.

Based on the recent development of the information state and the probabilistic approaches, [18] classifies the approaches in four categories:

(1) Finite-state and frame-based approaches,

(2) Information state and the probabilistic approaches,

(3) Plan-based approaches, and

(4) Collaborative agent-based approaches.

Each of these approaches makes different assumptions about the nature of the interaction; each has its own advantages and disadvantages. In this section, we briefly introduce each of these technologies, to provide the background for the rest of the presentation. A more in-depth review of these technologies falls outside the scope of this paper [61].

Finite state

In a finite-state dialog manager, the flow of the interaction is described via a finite-state automaton.

At each point in the dialog, the system is in a certain state (each state typically corresponds to a system prompt). In each state, the system expects a number of possible responses from the user; based on the received response, the system transitions to a new state.

To develop a dialog manager for a new application, the system author must construct the corresponding finite state automaton. In theory, the finite-state automaton representation is flexible enough to capture any type of interaction.

In practice, this approach is best suited only for implementing relatively simple systems that retain the initiative throughout the conversation. In these cases, the finite-state automaton representation is very easy to develop, interpret, and maintain.

However, the finite-state representation does not scale well for more complex applications or interactions. For instance, in a mixed-initiative system (where the user is also allowed to direct and shift the focus of the conversation), the number of transitions in the finite-state automaton grows very large; the representation becomes difficult to handle.

One representative example of this approach is the CSLU dialog management toolkit.

In this kind of systems, a finite state machine represents the dialogue. This means that each state transition must be codified in the system. The change of state occurs when the user provides information which the system is expecting. This information is generally a short phrase or even single words.

By knowing exactly what question you are responding to on each moment of the interaction, it is possible to adjust speech recognition for better performance. Using the same method, natural language understanding can also be benefited.

It is theoretically possible to create finite state machines where on each state there are transitions that match the user's possible responses. In practice, however, that would mean an explosion in the number of states which makes the construction of such systems impossible.

These systems do not offer much freedom to the user because the answers must be given in the pre-determined order, and moreover, the user must not answer more than he was asked. Systems that control dialogue in such a way are considered single initiative or system initiative.

Another problem is related to verifying information strategies. Here, the verification is done through explicit confirmation, which can lead to very long dialogues.

Form filling

Another dialog management technology, especially useful in information access domains is form-filling (also known as slot-filling). In this case the basis for representing the system's interaction task is the form (or frame). A form consists of a collection of slots, or pieces of information to be collected from the user.

For instance, in a train schedule information system, the slots might be the departure and arrival city, the travel date and time. Each slot has an associated system prompt that will be used to request the corresponding information from the user.

Typically, an action is associated with each form, for instance access to the schedule database in the train system. The system guides the dialog such as to collect the required slots from the user (some of the slots might be optional); the user can also take the initiative and provide information about slots that the system has not yet asked about. Once all the desired information is provided, the system performs the action associated with the form.

The system's interaction task may be represented as a collection of chained forms, with a specified logic for transitioning between these forms. In comparison with the finite-state representation, the form-filling approach makes stronger assumptions about the nature of the interaction task, and in the process allows system authors to more easily specify it.

As we have mentioned before, this approach is well-suited in information access domains where the user provides some information to the system, and the system accesses a database or performs an action based on this information.

However, the approach cannot be easily used to construct systems in domains with different interaction styles: tutoring, guidance, message delivery, etc.

Representative examples of this approach are the industry standard Voice-XML and the Phillips' SpeechMania system [61].

Information state

A third dialog management approach that has recently received a lot of attention and wide adoption in the research community is information-state-update (ISU). In this approach the

interaction flow is modelled by a set of update rules that fires based on the perceived user input and modify the system's state.

The system state (also known as information-state) is a data structure that contains information accumulated throughout the discourse. The information-state-update approach allows for a high of flexibility in managing the interaction.

Different ISU systems can capture different information in the state, and implement different linguistic theories of discourse in the state-update rules.

A potential drawback of this approach is that, as the set of update rules increases, interactions between these rules and their overall effects become more difficult to anticipate.

Representative examples of the ISU approach include the TrindiKit dialog move engine [62] and DIPPER [63].

In these systems, the context of the dialog is kept, called information state, which identifies anything that occurred in the dialogue so far and also guides the decisions of the dialog manager.

A dialogue system based on state information must contain [59]:

- Description of the information state - a description of the components that comprise the information state.
- Formal representation - formal identification of the components described in the information state.
- Dialogue moves - which can be viewed as external events that trigger the update of the information state.
- Update Rules - define when and how to update the information state and usually activated by the dialog moves.
- Update Strategy - determines which rules to apply and in what order.

In 2006, Traum and co-workers extended the original idea of the information state to develop a multi-layer dialogue model, each layer contains an information state representing the current status of that layer and a set of dialogue acts corresponding to the well-defined changes to the information state [18].

Another extension the information state approaches is to use probabilistic techniques such as Markov Decision Process (MDP) or a Partially Observable Markov Decision Process (POMDP). The idea is to dynamically allow changing of the dialogue strategy and the actions of dialogue systems based on optimizing some kinds of rewards or costs given the current state [18]. A Spoken Dialog POMDP based system is presented in [64] .

Plan Based

The fourth dialog management technology we have mentioned are plan-based approaches. These approaches are based on the plan-based theories of communicative action and dialogue. The theories claim that the speaker's speech act is part of a plan and that it is the listener's job to identify and respond appropriately to this plan [18].

In this case, the system models the goals of the conversation, and uses a planner to guide the dialog along a path towards these goals. These systems reason about user intentions, and model relationships between goals and sub-goals in the conversation, and the conversational means for achieving these goals. As a consequence, they require more expertise from the system developer, but can enable the development of more complex interactive systems. Examples include the TRAINS and TRIPS systems.

The RavenClaw dialog manager bares most similarities to this last class of systems, following essentially a hierarchical plan-based approach [61].

"Plan-based approaches have been criticized on practical and theoretical grounds. For example, the processes of plan-recognition and planning are combinatorically intractable in the worst case, and in some cases they are even undecidable. These approaches also lack of a sound theoretical basis. There is often no specification of what the system should do, for example, in terms of the kinds of dialogue phenomena and properties the framework can handle or what the various constructs like plans, goals, etc. are." [18].

Collaborative agent-based approaches [18].

Collaborative approaches or agent-based dialogue management approaches are based on viewing dialogues as a collaborative process between intelligent agents. Both agents work together to achieve a mutual understanding of the dialogue. The motivations that this joint activity places on both agents motivates discourse phenomena such as confirmation and clarification - which are also evident in human to human conversations [18].

Unlike the dialogue grammars and plan-based approaches which concentrate on the structure of the task, the collaborative approaches try to capture the motivations behind a dialogue and the mechanisms of dialogue itself. The beliefs of at least two participants will be explicitly modelled. A proposed goal, which is accepted by the other partner(s), will become part of the shared belief and the partners will work cooperatively to achieve this goal.

Several classes of these approaches have been developed using theorem proving, distributed architectures, and conversational agents. In some approaches, agents collaborate to build a mutual model of conversation and shared belief using a set of domain dependent and independent speech acts. Others extend Bratman's Beliefs Desires Intentions (BDI) agent architecture. In it, actions in the world affect agent's beliefs and the agent can reason about

its beliefs and thus formulate desires and intentions. Various recent dialogue management frameworks have been following the collaborative approaches such as COLLAGEN and TRIPS. The advantages of the collaborative approaches are the ability to deal with more complex dialogues that involve collaborative problem solving, negotiation, and so on. But the approaches require much more complex resources and processing than the dialogue grammars and plan-based approaches [18].

Multi-strategy systems

There are proposals to use multiple dialogue strategies in the same Spoken Dialogue System (SDS). In that case, the Dialog Manager would be responsible for changing the strategy according to various circumstances. The general idea behind it is to apprehend the best that each strategy has to offer and thus make man-machine interaction as natural as possible [65].

This new manager will be integrated into Queen's Communicator dialog manager.

The RavenClaw dialog management framework

A state-of-the-art Spoken Dialog Management Frameworks is CMU RavenClaw [61].

RavenClaw is a two-tier dialog management framework that enforces a clear separation between the domain-dependent and the domain-independent aspects of the dialog control logic. The domain-specific aspects are captured by the dialog task specification, essentially a hierarchical-plan for the interaction, provided by the system author. A reusable, domain-independent dialog engine manages the conversation by executing the given dialog task specification. In the process, the dialog engine also contributes a basic set of domain-independent conversational strategies such as error handling, timing and turn-taking behaviors, and a variety of other universal dialog mechanisms, such as help, repeat, cancel, suspend/resume, quit, start-over, etc. [61].

Developing a new dialog manager using the RavenClaw framework therefore amounts to writing a new dialog task specification. More specifically, a dialog task specification consists of a tree of dialog agents, where each agent is responsible for handling a subpart of the interaction.

The dialog engine algorithms are centered on two data-structures: a dialog stack, which captures the discourse structure at runtime, and an expectation agenda, which captures what the system expects to hear from the user in any given turn. The dialog is controlled by interleaving Execution Phases with Input Phases [61].

The error handling architecture in the RavenClaw dialog management framework subsumes two main components:

(1) A set of error recovery strategies, and

(2) An error handling decision process that triggers these strategies at the appropriate time.

Based on the type of problem they address, the error recovery strategies in the RavenClaw dialog management framework can be divided into two groups [61]:

(1) Strategies for handling potential misunderstandings and

(2) Strategies for handling non-understandings.

Dialogue Systems & Development Tools

Very representative examples of state-of-the-art Development Tools for SDS and of SDS systems/applications are presented in this section. The more attention/detail of description of some systems is directly related to its relevance in the current state-of-the-art, i.e., more influential tools and systems are given more attention.

Olympus dialog system infrastructure

Olympus is a dialog system infrastructure that, like RavenClaw, has its origins in the earlier CMU Communicator project [61]. The CMU Communicator Project was a Defense Advanced Research Projects Agency (DARPA) project with had the objective of promoting the development of SDS in various research institutes. The proposed scenario involved booking hotels and air tickets, and their confirmation by sending an e-mail. Project participants were: AT&T, BBN Technologies, Carnegie Mellon University, University of Colorado, IBM, Lucent Bell Labs, MIT, MITRE and SRI International.

At the high-level, Olympus implements a classical dialog system architecture (Figure 5). Each component is implemented as a separate process that connects to a centralized traffic router – the Galaxy hub. The messages are sent through the hub, which forwards them to the appropriate destination. The routing logic is described by a configuration script.

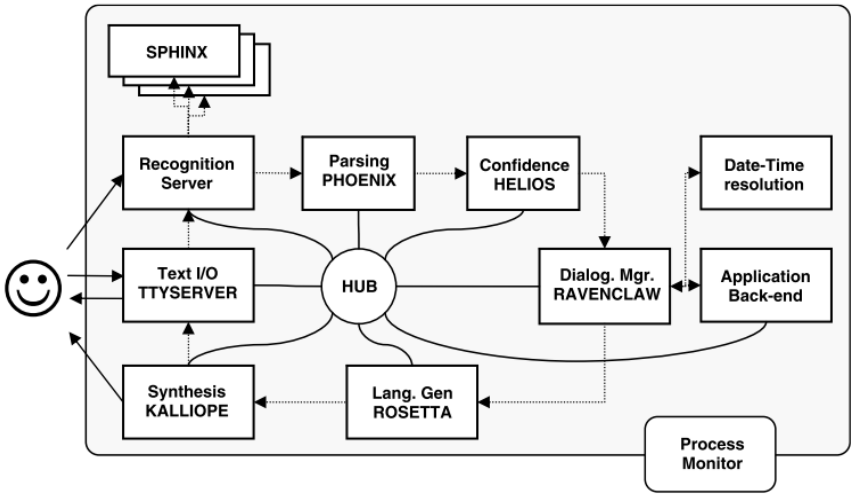

Figure 5 - The Olympus/RavenClaw dialog system architecture: a detailed view, from [61].

For recognition, Olympus uses the Sphinx decoding engine. A recognition server component captures the audio stream (typically from the sound-card), forwards it to a set of parallel recognition engines, and collects the corresponding recognition results. The top-level recognition hypotheses (one from each engine) are then forwarded to the language understanding component. Currently, Sphinx-II (semi-continuous HMM recognition) and Sphinx-III (continuous HMM recognition) engines are available and can be used in conjunction with the recognition server. Additionally, a DTMF (touch-tone) decoder is also available as a recognition engine.

The RavenClaw/Olympus systems described in the next section use two parallel Sphinx-II recognizers: one configured with acoustic models trained using male speech and the other configured with acoustic models trained using female speech. Other parallel decoder configurations can also be created and used.

Language understanding is implemented via Phoenix, a robust semantic parser. Phoenix uses a semantic hand-written grammar to parse the incoming set of recognition hypotheses (one or more parses can be generated for each hypothesis). The semantic grammar is constructed by concatenating a set of reusable grammar rules that capture domain-independent constructs like [Yes], [No], [Help], [Repeat], [Number], etc., with a set of domain-specific grammar rules authored by the system developer. For each recognition hypothesis the output of the parser consists of a sequence of slots containing the concepts extracted from the utterance.

From Phoenix, the set of parsed hypotheses is passed to Helios, the confidence annotation component. Helios uses features from different knowledge sources in the system (e.g., recognition, understanding, dialog, etc.) to compute a confidence score for each parsed

hypothesis. This score reflects the probability of correct understanding, i.e. how much the system trusts that the current semantic interpretation corresponds to the user's expressed intent. The hypothesis with the highest confidence score is then forwarded to the dialog manager.

The next component in the chain is the RavenClaw-based dialog manager. The dialog manager integrates the semantic input in the current discourse context, and decides which action the system should engage in next. In the process, the dialog manager may consult/exchange information with a number of other domain-specific agents, such as an application-specific back-end.

The semantic output from the dialog manager is sent to the Rosetta language generation component, which creates the corresponding surface form. Rosetta supports template-based language generation. Like the grammar, the set of language generation templates is assembled by concatenating a set of predefined, domain-independent templates, with a set of manually authored task-specific templates.

Finally, the prompts are synthesized by the Kalliope speech synthesis module. Kalliope can be configured to use a variety of speech synthesis engines: Festival, which is an open-source speech synthesis system, as well as Cepstral Theta and Cepstral Swift, which are commercial solutions. Kalliope supports both open-domain (e.g. diphone) and limited-domain (e.g. unit selection) voices. The SSML markup language is also supported.

Several applications were developed using OLYMPUS, such as [61]: RoomLine, a telephone-based mixed-initiative spoken dialog system that provides conference room schedule information and allows users to make room reservations; Let's Go! Public Bus Information System, a telephone-based system that provides access to bus route and schedule information; **LARRI,** or the Language Based Retrieval of Repair Information **system is a multi-modal system for support of maintenance and repair activities for F/A-18 aircraft mechanics**; TeamTalk, a multi-participant spoken language interface that facilitates communication between a human operator and a team of robots. The system operates in a multi-robot-assisted treasure-hunt domain.

TrindiKit

The description on this section is based on [59, 62].

TrindiKit is a toolkit for building and experimenting with dialogue move engines and information states. The term information state (as explained before) refers to the information stored internally by an agent, in this case a dialogue system. One of TrindiKit main functions comes with the dialogue move engine, or DME, which updates the information state on the basis of observed Dialogue Moves and selects appropriate moves to be performed.

Apart from proposing general system architecture (Figure 6), TrindiKit also specifies formats for defining information states, update rules, dialogue moves, and associated algorithms. It also provides a set of tools for experimenting with different formalizations of implementations of information states, rules, and algorithms.

To build a dialogue move engine, one needs to provide definitions of update rules, moves and algorithms, as well as the internal structure of the information state. The DME forms the core of a complete dialogue system. TrindiKit provides simple interpretation, generation and input/output modules in order to simulate an end-to-end dialogue system.

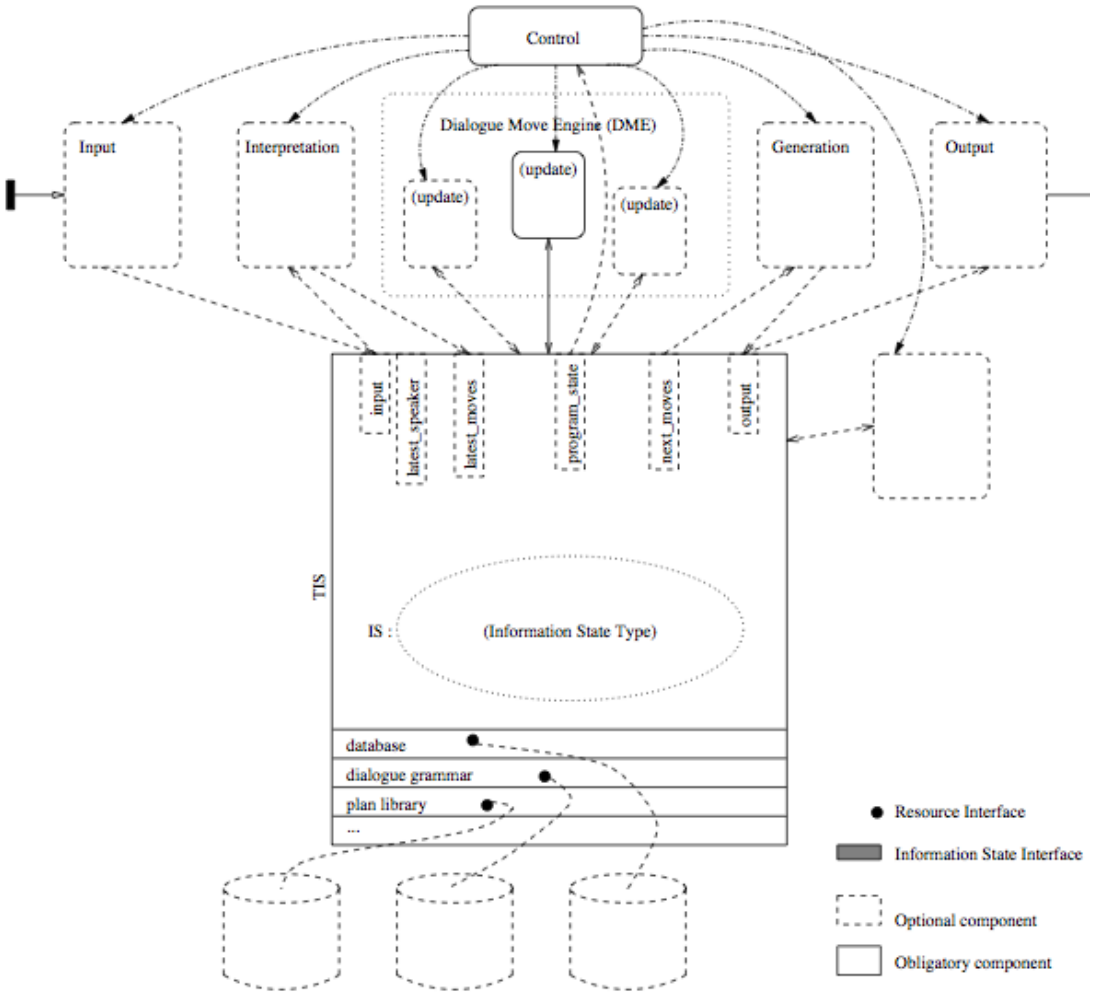

Figure 6 - TrindiKit's System Architecture Sample Setup

TrindiKit defines information states using a rich set of datatypes, including records, stacks and queues. It allows developers to define specific information states, tailored to a particular theory or a special task. An information state is normally defined as a recursive structure of the form Name:Type, where Name is an identifier, and Type a datatype.

In order to achieve this, the general architecture (Figure 6) on TrindiKit is composed by the following standard components:

1. the Total Information State (TIS) , consisting of
    a. the Information State (IS) variable
    b. module interface variables
    c. resource interface variables
2. modules, operating according to module algorithms
3. the Dialogue Move Engine (DME), consisting of one or more modules; the DME is responsible for updating the IS based on observed moves, and selecting moves to be performed by the system.
4. a controller, wiring together the other modules, either in sequence or through some asynchronous mechanism.
5. resources, such as databases, etc.

The Total Information State is accessed by modules through conditions and operations. The types of the various components of the TIS determine which conditions and operations are available.

DIPPER

The description on this section is adapted from [63].

The DIPPER architecture is a collection of software agents for **prototyping spoken dialogue systems**. Implemented on top of the Open Agent Architecture (OAA), it comprises agents for speech input and output, dialogue management, and further supporting agents.

This framework is based on TrindiKit. However, the authors argue that on TrindiKit, what should be a transparent operation is often obscured by its complexity. Nonetheless, the dialogue management component of DIPPER borrows many of the core ideas of the TrindiKit, but is stripped down to the essentials, uses a revised update language (independent of Prolog), and is more tightly integrated with OAA.

They believe that the resulting formalism offers several advantages for developing flexible spoken dialogue systems. In comparison to TrindiKit, they state that DIPPER provides a transparent and elegant way of declaring update rules independent of any particular programming language, and with the ability to use arbitrary procedural attachment via OAA.

Architecture

A prototypical spoken dialogue system built on top of OAA consists of an agent for speech recognition, an agent for dialogue management, an agent for speech synthesis, and several supporting agents for specific tasks such as parsing, semantic interpretation, and generation.

The current collection of DIPPER agents consists of the following: (1) agents for input/output modalities, (2) agents for the dialogue move engine, and (3) supporting agents.

An example of an input agent in DIPPER is an agent for speech recognition (from Nuance Software). It can be used in two different modes: continuous speech recognition, calling the functionality "apply effects"(+Effects) and thereby updating the information state of the dialogue; and in callback mode, where the functionality recognize (+Grammar,+Time,-Input) starts recognition using the speech grammar and returns Input, within a time specified by Time.

In DIPPER, the **dialogue manager** is implemented as two cooperating OAA agents: the dialogue move engine (DME), and a DME server. The DME's function is to deal with input from other agents (normally the input modalities, such as speech recognition), update its internal state, and call other agents. The DME server is an agent mediating between the DME and other agents. It collects requests submitted by the DME, waits for the results, and posts these back to the DME. The DME server enables the DME to manage information-state updates in an asynchronous way.

To summarize the functionality of the DME, there are three ways it is able to communicate with other agents in a dialogue system: (1) agents can call the DME agent directly; (2) the DME agent can call other agents directly, in particular if it is not interested in the results of those requests; (3) the DME agent can use the DME server as a mediating agent, normally when the results are needed for updating the information state of the DME.

In addition to the input/output and DME agents, DIPPER also contains various support agents. DIPPER provides a set of agents to deal with natural language understanding, based on Discourse Representation Theory. There is also an ambiguity resolution agent that resolves underspecified DRSs into fully resolved DRSs, and there is an inference agent that checks consistency of DRSs, using standard first-order theorem proving techniques, including the theorem prover SPASS and the model builder MACE. DIPPER also includes a high-level dialogue planning component using O-Plan which can be used to build domain-specific content plans.

CARDIAC SDS

CARDIAC is an agent-based spoken dialogue system that conducts health monitoring interviews with chronic heart failure patients using natural language [66].

Carl Spoken Language Interface

Lopes et al. [67] describes an integrated set of capabilities developed to support knowledge acquisition in a mobile robot through spoken language interaction with its users. The robot (Carl) is a prototype of an intelligent service robot, designed and developed having in mind hosting tasks in a building or event.

Input is provided by modalities such as a touch screen monitor, a directional microphone array and an on board camera, all equipped in the robot.

Carl's software architecture utilizes OAA for interconnecting the agents. For instances, speech processing is handled by an ASR agent and touch is controlled by a Graphical and Touch Interface (GTI) agent.

In order for the robot to accept instructions or acquire knowledge from human interlocutors though spoken language, in addition to the ASR agent, a Natural Language Understanding (NLU) agent is also implemented. This way, after the de-codification of voice inputs to sequences of words by the ASR, the NLU processes those sequences in order to extract the maximum information possible. This is done in two stages: first a syntactic structure is built and then the semantic information is extracted.

The communication is modelled as the exchange of messages. The currently supported set of message types in Carl's Human-Robot Communication Language (HRCL) includes: register, achieve, tell, ask, ask if, thanks, bye and sleep.

The Manager agent uses the information state (IS) approach to handle dialogues and control the robot. Besides the information state description, the dialogue manager is also composed by events (external occurrences leading to an information state update); IS update rules (defines when and how to update the IS); action selection (defines which action to perform next); and control (decides which update rules are applied and selects the next action).

The Manager also contains the Knowledge Acquisition and Management (KAM) module. The definition of their KA language is based on typical definitions of semantic networks and on class and object diagrams of UML.

The Navigation agent handles the robot's general perception and action. It is based on Saphira and ARIA, the software interface for Pioneer robots.

Multimodal Dialog Systems

Previous research work - and previous subsection - has been focusing on spoken dialogue systems, which are defined as computer systems that human interact on a turn-by-turn basic and in which spoken natural language interface plays an important part in the communication [18].

Recently, it has been extended to multimodal dialogue systems, which are dialogue systems that process two or more combined user input modes - such as speech, pen, touch, manual gestures, gaze, and head and body movements - in a coordinated manner with multimedia system output [18].

The general idea of the information state approaches is being used for the development of multimodal dialogue systems such as Virtual Music Center, MATCH system for multimodal access to city help, Immersive Virtual Worlds [18].

An often cited report on this subject is [68].

Fission of Output Modalities

This section is strongly based in [69].

In multimodal interactive systems, fission is the process of realizing an abstract message through output on some combination of the available channels [18].

A multimodal presentation is composed of a set of output (modality, medium) pairs built by redundancy or complementarity properties. For example, an incoming call on a mobile phone may be expressed through a multimodal presentation composed of two pairs. A first pair ("ringing modality", "speaker medium") indicates a phone call while a second pair ("text modality", "screen medium") presents the caller's identity [70].

In general, the main tasks of a fission module fall into three categories[18]:

- The content to be included in the presentation must be selected and arranged into an overall structure (content selection and structuring);
- The particular output that is to be realised in each of the available modalities must be specified (modality selection);
- The output on each of the channels should be coordinated so that the resulting output forms a coherent presentation (output coordination).

Content Selection and structuring

Content selection and structuring together constitute the task of designing the overall structure of a presentation. Since multimodal presentations generally follow structuring principles similar to those used in text, most multimodal generation systems use techniques derived from text planning.

Early research in language generation showed that producing natural-sounding multi-sentential texts required the ability to select and organize content according to rules governing discourse structure and coherence. There is a growing consensus among researchers that at least three types of structure are needed in computational models of discourse:

- Intentional structure - describes the roles that utterances (the term "utterance" in normally used in spoken language rather than "sentence") play in the speaker's communicative plan to achieve desired effects on the hearer's mental state or the conversational record;
- Informational structure - consists of the semantic relationships between the information conveyed by successive utterances;
- Attentional structure - contains information about objects, properties, relations, and discourse intentions that are most salient at any given point in the discourse.

In some cases, the content selection and structuring is done by another process or by the user, which means, the content that is to be presented is determined before the fission process begins. However, in other cases, selecting and structuring the content does form part of the fission process.

Approaches:

In order to select and structure the content, the most used approaches are schema-based or plan-based.

The notion of a schema was first proposed by McKeown [113], in 1985, in the context of text generation. A schema encodes a standard pattern of discourse by means of rhetorical predicates that reflect the function each utterance plays in the text. By associating each rhetorical predicate with an access function for an underlying knowledge base, these schemas can be used to guide both the selection of content and its organization into a coherent text to achieve a given communicative goal.

Researchers have applied techniques from AI planning research to the problem of constructing discourse plans that explicitly link communicative intentions with communicative actions and the information that can be used in their achievement. Text planning generally makes use of plan operators' discourse action descriptions that encode knowledge about the ways in which information can be combined to achieve communicative intentions.

Plan operators may include parameters such as: Effect(s) (the communicative goal(s) the operator is intended to achieve); Preconditions (the conditions that must hold for an act to successfully execute, e.g., it may be the case that the hearer must hold certain beliefs or have certain goals for a particular discourse strategy to be effective); Constraints (the specifications of the knowledge resources needed by the discourse strategy); Subplan (optionally, a sequence of steps that implement the discourse strategy).

Modality Selection

The objective of modality selection can be resumed by the following sentence: **Given a set of data and a set of media, find a media combination that conveys all data effectively in a given situation.**

To perform modality selection, some or all of the following forms of knowledge may be used:

1. The characteristics of the available output modalities.
2. The characteristics of the information to be presented.
3. The communicative goals of the presenter.
4. The characteristics of the user.
5. The task to be performed by the user.
6. Any limitations on available resources.

Modality characteristics - In most implemented systems, the available modalities are characterized in terms of either the (application-specific) types of information that they can present, or the perceptual tasks that they permit.

Data characteristics - Aspects of the data that can influence the modality-selection process include: dimensionality, transience, urgency, density, and “volume” (how much information there is to present). Several of these characteristics interact with other types of knowledge; for example, the urgency of a piece of data is determined largely by the user’s task or the presenter’s goals, while the amount of data that constitutes “too much” is a user-dependent feature.

User characteristics - While many multimodal presentation systems incorporate a user model, very few make use of it at the point of modality selection.

User task and presenter goals - In most multimodal presentation systems, the main goal is to present some information to the user, and the user’s task is essentially to understand the information that is presented. In other words, the presentation is designed only to perform information transfer, rather than to engage in any sort of dialogue. This means that it is often difficult to distinguish the user’s task from the presenter’s goals in this context. In most systems that take into account these factors, modality selection and content selection and structuring take place at the same time; in fact, the particular content structure often determines the modalities.

Resource limitations - The main form of resource limitation that can affect modality selection is the physical size of the display device. Many systems combine modality selection with physical layout in the presentation-planning process, so if the first-choice combination of modalities cannot be made to fit into the screen space available, the planner can backtrack over other possible modalities until a combination is found that can fit.

In order to execute the selection process, existing systems take a variety of approaches. Some of these approaches are:

- Composition - (1) The components of the output specification are grouped into compatible sets. (2) For each grouping, the system selects the graphical presentation techniques that can express that grouping. The techniques are then ranked by their effectiveness. (3) The system tries to combine the selected graphical primitives, using predefined composition operators. If the top-choice candidates cannot be combined using the operators, the remaining candidates are tried in order until one is successful.
- Rules - Many systems use rules to allocate the components of the presentation among the modalities (example of a rule: "If there is a large amount of information to present, do not use a transient modality")
- Plan-based approaches - In the systems that use a plan-based approach to content selection and structuring, modality selection takes place as a side effect of selecting among presentation strategies, and the necessary knowledge is encoded in the strategies themselves.
- Competing and cooperative agents - A hierarchical system of competing and cooperative agents to plan its presentations. An individual agent is created to attempt to realise each piece of information in each modality attached to that information.

Output Coordination

Output coordination is the task of ensuring that the combined output from the individual generators amounts to a coherent presentation. Coordination may take several forms, depending on the particular modalities that are used and the emphasis of the presentation system. Some approaches are the following:

- Physical layout - When more than one visually-presented modality is used the individual components of the presentation must be laid out;
- Temporal coordination - If the presentation includes dynamic modalities such as speech or animation, these presentation events must be coordinated in time;
- Referring expressions - Some systems further coordinate their output by producing multimodal and cross-modal referring expressions.

Current models and implementations

WWHT Model [70]

The WWHT conceptual model [70] is based on four concepts ("What", "Which", "How", "Then") describing the life cycle of an adapted multimodal presentation:

- What is the information to present?
- Which modality(ies) should we use to present this information?
- How to present the information using this(ese) modality(ies)?
- Then, how to handle the evolution of the resulting presentation?

The three first concepts (What, Which and How) refer to the build process of a multimodal presentation (Figure 7). This build process can be divided into three steps.

The first step (What) called the "semantic fission" decomposes the semantic information issued from the dialog controller into elementary information.

The second step (Which) allocates a multimodal presentation to express this information. For each elementary information, an "election" of the best (modality, medium) pairs according to the interaction context state is done.

All these elements define a multimodal presentation expressing the initial information.

The last step (How) instantiates the elected multimodal presentation. The "instantiation" process selects concrete content to express through the selected modalities and sets presentation attributes (modalities attributes, spatial and temporal parameters, etc.).

Finally the "rendering engine" presents the multimodal presentation to the user [70].

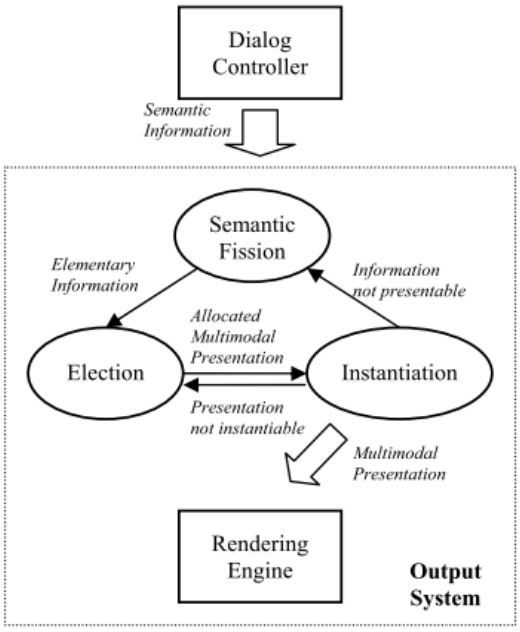

Figure 7. Presentation of a semantic information.

Figure 7 – Presentation of semantic information (from [70])

MOSTe Tool

A tool called MOSTe (Multimodal Output Specification Tool) [70] has been implemented in order to make easier the specification process.

This tool is composed of four editors (component editor, context editor, information editor and behavior editor) corresponding to each task of the specification. MOSTe allows the reuse of the outputs specification during the design process.

Output

Various output modalities can be used to present the information content from the fission module such as: speech, text, 2D/3D graphics, avatar, haptics, and so on [18].

Popular combinations of the output modalities are [18]:

(1) Graphics and avatar,

(2) Speech and graphics,

(3) Text and graphics,

(4) Speech and avatar,

(5) Speech, text, and graphics,

(6) Text, speech, graphics, and animation,

(7) Graphics and haptics,

(8) Speech and gesture.

The combinations are marked in the following table:

Table 5 - Combinations of output modalities

Combination	Graphics	Avatar	speech	text	haptics	animation	gesture
1	X	X					
2	X		X				
3	X			X			
4		X	X				
5	X		X	X			
6	X		X	X		X	
7	X				X		
8			X				X

In this section we review some of what is presently available to be used for output in multimodal interfaces and the most advanced ways of using them. State-of-the-art for the several output modalities is considered outside the scope of this report. We also don't address the common Text output and output part of the widespread WIMP interfaces (Windows, Icons, Menus, Pointer).

Graphics and Text

The graphical and textual outputs are typically presented in two-dimensional display screens, with display resolutions able to describe generic information. It represents the information and actions available to a user through graphical icons and visual indicators. The actions are usually performed through direct manipulation of the graphical elements. This way, users interact with information by manipulating visual widgets that allow for interactions appropriate to the kind of data they hold.

Secondary notations are amply used. A secondary notation is defined as "visual cues which are not part of formal notation". Properties like position, indentation, color, symmetry, when used to convey information, are secondary notation. A typical example of secondary notation is syntax highlighting of programming code: the colors are not part of the code semantics, but help the programmer to visualize its meaning.

A visual interface is a constant element in almost every interaction with an electronic device, usually being the primary output channel. As we have seen, graphics and text are present in 6 of the 8 popular interface combinations.

Speech output

As for input, speech is also one of the commonly used modality in multimodal systems. It is part of 5 of the 8 popular combinations mentioned before.

Sun MicroSystems recommendations [21, 26] are presented in Table 6 – Recommendations regarding the use of speech output.

Speech (and audio) output	
Appropriate when . . .	**Inappropriate when . . .**
• there is no other output mode available • no other output mode is practical in the device context (e.g., there is no visual display) • no other output mode is practical in the user's knowledge context (e.g., users are illiterate) • no other output mode is practical in the user's physical context (e.g., user's eyes are physically disabled) • the task requires the user's eyes to be looking at something other than a visual display (e.g., driving, maintenance and repair)	• large quantities of information must be presented to the user • confidentiality is important (e.g., for privacy or security) • the environment is very noisy • tasks are easier with other output modes (e.g., comparing data items is a significant aspect of the task)

Table 6 – Recommendations regarding the use of speech output.

To use speech output one needs a Text-to-Speech (TTS) Engine for the target language(s), a way of interfacing the TTS to the Fission module.

TTS available for Portuguese are available from Microsoft, Voice Interaction, Loquendo and Nuance.

An adequate way of sending information to the TTS is by using a Speech Markup Language.

Speech Synthesis Markup Language (SSML) is an XML-based markup language for speech synthesis applications. In its version 1.1, is a Proposed Recommendation of W3C since 23 February 2010 [71]. SSML is based on the Java Speech Markup Language (JSML) developed by Sun Microsystems. It covers virtually all aspects of synthesis, although some areas have been left unspecified, so each vendor accepts a different variant of the language. Also, in the absence of markup, the synthesizer is expected to do its own interpretation of the text.

For desktop applications, other markup languages are popular, including Apple's embedded speech commands, and Microsoft's SAPI Text to speech (TTS) markup, also an XML language.

Large Screen

As technology prices have fallen in recent years, large screen displays have become increasingly prevalent. However, currently many of these displays are broadcast only information sources and there is considerable interest in research about ways that these large public situated displays can be adapted for better human interaction.

Haptic

A haptic output is a tactile feedback technology which takes advantage of the sense of touch by applying forces, vibrations, or motions to the user. This mechanical stimulation can be used to assist in the creation of virtual objects in a computer simulation, to control such virtual objects, and to enhance the remote control of machines and devices. Haptic devices may incorporate tactile sensors that measure forces exerted by the user on the interface.

Since haptic feedback cannot convey much information, it is typically used as a complementary modality, giving users feedback about the actions they are performing or alerting them to certain events (e.g., receiving a phone call). It is particularly important when environmental conditions compromise other interaction interfaces, e.g., it is difficult to hear audio outputs in a noisy environment.

The intensity of the haptic feedback is a key factor. If a haptic device (e.g. a phone) is held in the hand or it is on a hard surface (e.g. a table), it is very easy to detect the vibration. But if it is in the pocket of a coat or in a backpack, the situation changes drastically. This fact should be taken into account when designing haptic interfaces.

In a study performed by Ki-Uk et al. [Ki-Uk 09], they designed a haptic stylus interface for interacting with a touch screen. Results showed that haptic cues improved the performance of users, when compared to the visual-only interaction. It contributes to preciseness, and makes the user more comfortable and confident.

Development Tools and Languages

This section presents information regarding two important resources available to make possible development of multimodal interaction for an application: development tools – such as toolkits -, and specialized languages.

Development tools & Frameworks

Tools to develop Multimodal Interfaces include Workbenches, toolkits, APIs and Frameworks. The ones considered more relevant are summarized in this section.

According to [72] "There is currently few ready-to-use software solutions aimed at filling the gap between the design and specification stage and the implementation process of a functional system."

Many of the existing tools for the iterative design of multimodal systems have the following problems [72]:

(1) present a small or hardly extensible number of input devices,

(2) They are platform and technology dependent, or

(3) They do not provide a flexible prototyping environment for a large and heterogeneous number of research products (such as new device prototypes, new algorithms, etc.).

The description of several tools for multimodal interaction creation is presented in the following subsections. For the most recent and with code or runtimes available, the description is much more detailed.

ICON

ICON is a java input toolkit that allows interactive applications to achieve a high level of input adaptability. It natively supports several input devices. Devices can be added to the toolkit using JNI, the low-level Java Native Interface allowing integration with programs written in C. (Information extracted from [72]).

ICARE

ICARE is a component-based platform for building multimodal applications. This solution defines a new component model based on Java Beans, and requires all components to be written in Java. The platform is not easily extensible, produces non-reusable components, and also requires additional programming effort for integrating new devices or features. (Information extracted from [72]).

CrossWeaver

CrossWeaver is a user interface design tool for planning multimodal applications. It allows the designer to informally prototype multimodal user interfaces. This prototyping tool supports a limited number of input and output modalities and is not suitable for integrating additional software components [72].

Exemplar

The goal of Exemplar is to enable users to focus on design thinking (how the interaction should work) rather than algorithm tinkering (how the sensor signal processing works). Exemplar frames the design of sensor-based interactions as the activity of performing the actions that the sensor should recognize. This work provides an Eclipse based authoring environment which offers direct manipulation of live sensor data [72].

OpenInterface

The OpenInterface project is a STREP (Specific Targeted Research Project) project of the European IST Framework 6 funded by the EC. It was a multidisciplinary project which involved (ended in May 2009) academic and industrial partners from different areas (human-computer interfaces, designers, software Computer engineers, etc.). This project was dedicated to multimodal interaction in order to meet the wide range of possibilities for interaction modalities, thus going beyond the traditional WIMP (screen-keyboard-mouse). Various everyday objects may participate in this interaction (e.g. a table with capabilities for viewing and response to touch) and users can switch modalities depending on the context (street running, home, car driving, etc.)

OpenInterface Kernel is a generic runtime platform for integrating heterogeneous code (e.g. device drivers, applications, algorithms, etc.) by means of non-intrusive techniques and with minimal programming effort, while achieving exploitable runtime performances (e.g. low latency, low memory overhead) [72].

The first design tool of OpenInterface was OIDE, a design tool presented as a development environment for multimodal interaction built on top of OpenInterface runtime platform.

The limitations of OIDE have motivated the developers to providing an all-in-one prototyping workbench for multimodal applications development, SKEMMI. It supports a multi-level interaction design and allows composition and modification of running applications through techniques such as design-by-demonstration or direct manipulation [72]. SKEMMI differs on the following main features, which are not addressed by OIDE: Support for components development; Support for reusability; Support for documentation; Runtime and Debug [72].

The runtime platform adopts an extensible modular architecture in which components are the base objects manipulated by the OpenInterface Platform.

Components can be implemented in virtually any language, we do not constrain to the use of a component model, and we strive for minimal programming efforts when integrating new components.

Therefore, not specifying an explicit model provides additional flexibility, i.e. the ability to implement/support various models for interactive systems (e.g. MVC, PAC, ARCH, etc.). Components are unaware of the platform in which they are running; therefore, programmers can use any preferred programming language and external tools, while only declaring interfaces [72].

Having components declare only their communication interface enforces the requirement of "independence". A component exports inputs and outputs to provide functionalities and services (e.g. Image display, device status) and imports inputs/outputs to request features provided by other components.

In order to declare interfaces, regardless of their implementation language, we define the XML-based CIDL description language (Component Interface Description Language - [72]).

In order to build a running application, project introduces the concept of Pipeline as an interconnection and configuration of components. It allows control over the components life-cycle and execution site (remote, local), provides low level (threshold, filter, etc.) and high level (multicast, synchronization, etc.) data flow control for building up a complex systems. A pipeline also supports dynamic reconfiguration of connections at runtime [72].

HephaisTK

HephaisTK is designed in the Java programming language, as a multi-platform toolkit [73].

HephaisTK [73] is intended to be a toolkit allowing rapid creation of multimodal interfaces, offering a predefined set of recognizers as well as the possibility to plug into the toolkit any other modality recognizer, as long as it complies with a given set of conditions, e.g. communication with the toolkit by means of the W3C EMMA language [73]. Its modular architecture allows developers to easily configure it according to their needs, and to plug new human-computer communications means recognizers.

HephaisTK (Figure 8) is designed to control various input recognizers, and, more importantly, user-machine dialog and fusion of modalities.

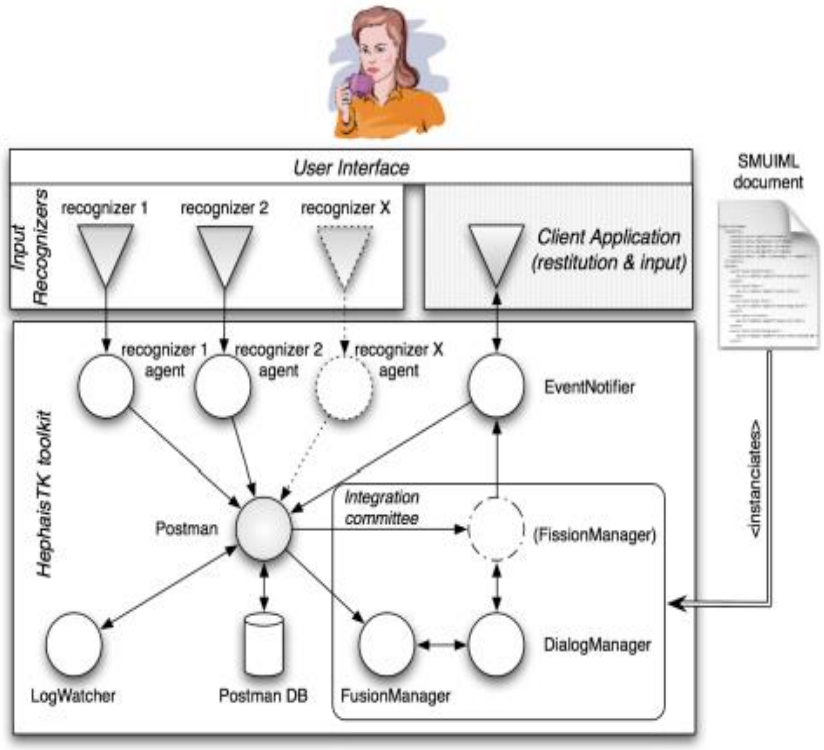

Figure 8 – HephaisTK Architecture.

A developer wishing to use HephaisTK to develop a multimodal application will have to provide two components: his application and a SMUIML script (Synchronized Multimodal User Interaction Markup Language).

The developer's application needs to import one class of HephaisTK. This class allows communication with the toolkit via Java listeners.

The SMUIML document is used by the toolkit for a number of tasks: first, the definition of the messages that will transit from the toolkit to the developer's application; second, the events coming from the input recognizers that will have to be taken into account by the toolkit; last, description of the overall dialog management.

HephaisTK also offer different fusion mechanisms to allow information from incoming recognizers to be extracted, and passed to potential client applications.

Free use of HephaisTK is possible through GPL licensing [73].

In its current state, HephaisTK is built upon a software agent system. Each time a new recognizer or synthesizer is plugged into the toolkit, an agent is dispatched to monitor it. Agents manage communication between the different parts of the framework, from the

input recognizer to the meaning extraction engines to the output modules. Agents are also used because of their ability to transit from one platform to another [73].

HephaisTK uses central blackboard architecture. A "postman" centralizes each message coming from the different input recognizers and stores it into a database.

Agents interested in a specific type of message can subscribe to the postman, which will accordingly redistribute received messages. Fusion of input modalities is achieved through meaning frames.

The toolkit manages fusion of modalities, as well as user-machine dialog, by means of an internal finite state machine paradigm; if the general dialog scheme is fixed, behavior of the fusion engine can be tuned by the developer to match the different CARE properties.

The fusion and dialog managers of HephaisTK are scripted by means of a SMUIML (Synchronized Multimodal User Interfaces Modelling Language) XML file (more information in section 0).

EPFL Framework for Rapid Multimodal Application Design

A problem that prevents spoken dialogue systems from broader use is the limited performance and reliability of current speech recognition and natural language understanding technologies. One of the research directions foreseen to overcome these limitations is the use of multimodal dialogue systems that exploit (besides speech) other interaction channels for the communication with the user. Within this perspective, the aim of this framework is to extend the EPFL (or Rapid Dialogue Prototyping Methodology (RDPM)) dialogue platform with multimodal capabilities.

Rapid Dialogue Prototyping Methodology [74]

The general idea behind the RDPM is to build upon the hypothesis that a large class of applications potentially interesting for the setup of interactive user-machine interfaces can be generically modelled in the following way: the general purpose of the application is to allow the users to select, within a potentially large set of targets, the one (or the ones) that best corresponds to the needs (search criteria) that are progressively expressed by the users during their interaction with the system.

Within this framework, a further assumption is made, the available targets can be individually described by sets of specific attribute:value pairs, and the goal of the interactive, dialogue based interface is then to provide the guidance that is required for the users to express the search criteria (i.e. the correct attribute:value pairs) leading to the selection of the desired targets.

The Rapid Dialogue Prototyping Methodology allows the production of dialogue models specific for a given application in a short time. In outline, the RDPM divides the design into the following **step**s:

(1) Producing a **task model** for the targeted application;

(2) Deriving an initial **dialogue model** from the obtained task model;

(3) Carrying out a series of Wizard-of-Oz experiments to iteratively improve the initial dialogue model.

The definition of the valid constrains (e.g. the list of available attributes and attribute combinations, as well as the possible associated values) is called the **task model**.

The **dialogue model** defines the types of interactions that are possible between the system and the user. In RDPM, the dialogue model consists of two main parts:

(1) Generic Dialogue Nodes (**GDNs**) and

(2) application-independent dialog flow management (dialog strategies).

For each attribute in the task model there is a GDN associated with it. Its role consists on performing the interaction with the user that is required to obtain a valid value for attribute. The term dialogue strategy refers here to the decision of the dialogue manager about the next step in the dialogue. The RDPM dialogue management handles dialogue strategies at two levels: local and global. Some local strategies refer to situations like requests for help or no input provided. As soon as the user provides a value compatible associated with a current GDN, control is handed back to the global dialogue manager where the global strategies are encoded. Some global strategies include confirmation strategies; incoherency strategies; a dialogue dead-end management strategy among others.

The Wizard of Oz Experiments allows the acquisition of experimental data about the behavior of the users when interacting with the system. Not yet implemented functionalities are simulated by a hidden human operator called the Wizard. In the experiments, the Wizard uses a specific interface to fulfill his task. That interface is generated automatically from the task and dialogue models.

Going Multimodal

To extend the method to be multimodal there is a need to cope with the problems of fusion and fission of modalities. As such, the first step is the creation of mGDNs (Multimodal GDNs). They follow the same principles as the already explained GDNs with some additional elements required for multimodal interaction like grammars for written and spoken natural language

input; a set of multimodal prompts to guide the user or the definition of roles for each GUI element.

The majority of the design principles for this framework revolve around the mGDN. These principles are mainly directed to resolve the multimodal fusion problem. The mGDN are the building blocks of the multimodal interface, with each type encapsulating a particular kind of interaction and providing various graphical layouts. In fact, the mGDN are the only interaction channel with the system available, that is, all inputs/outputs going to/coming from the system are managed by some mGDN. Each mGDN is multimodal, i.e. every mGDN gives the users the possibility to communicate using all the defined modalities. And at any given time, only one mGDN is operational in the interface (nonetheless, other may be active/ready, but not "running"). Another design principle is that during system design, only a limited number of modalities are to be taken into account.

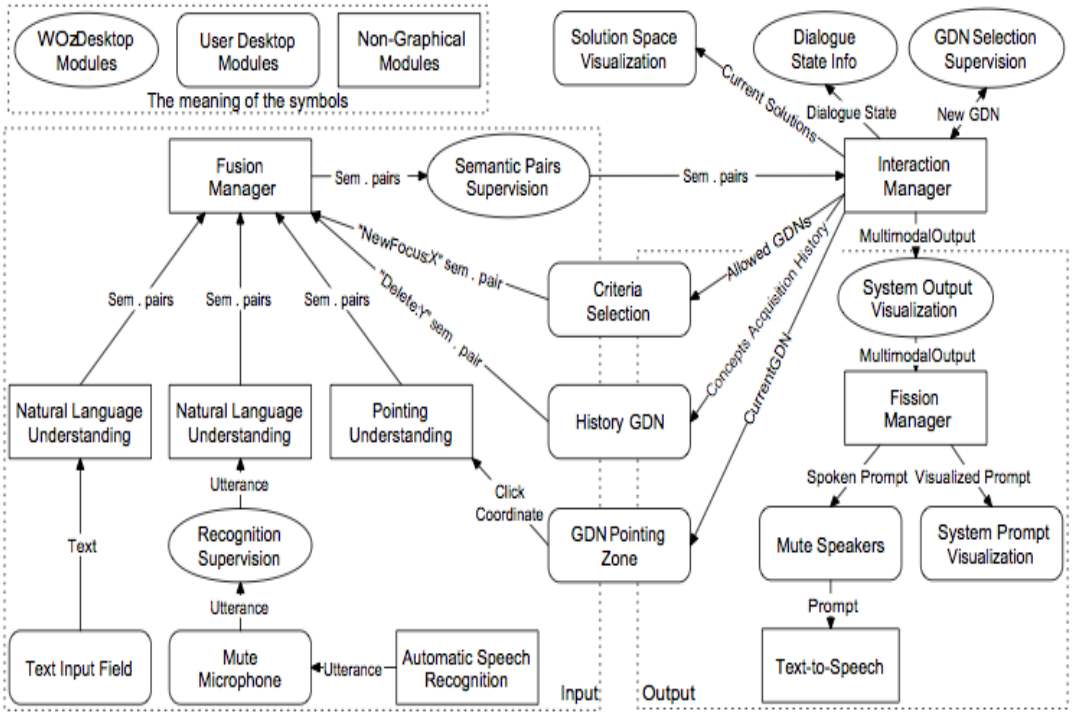

Figure 9 - EPFL Multimodal Framework Architecture [74].

The proposed module composition is depicted in Figure 9.

The **Interaction Manager** controls two groups of modules: the input and output modules.

The role of the Fusion Manager is to combine the semantic pairs from the different input sources (modalities).

The Text Input Field module allows the user to type in some text that is consequently translated by the NLU into semantic pairs. The same happens with the text produced by the ASR. Note that the ASR result might be corrected by Wizard's Recognition Supervision module. Possible values for the mGDN in focus are displayed in the GDN Pointing Zone. Mouse clicks are translated into semantic pairs by the Pointing Understanding module. The graphical modules History and Criteria Selection mGDNs work in a similar fashion except that they display only one GDN. Semantic pairs resulting from the fusion process are supervised by the Wizard in the Semantic Pair Supervision module and are then sent to the Interaction Manager.

The Interaction Manager processes the semantic pairs and selects the next GDN to be in focus (the decision can be modified by the Wizard in the GDN Selection Supervision). The dialogue state information is then updated, the Solution Space Visualization is modified and the multimodal output is issued by the Interaction Manager.

The output is sent by the Fission Manager to the System Prompt Visualization module that displays it on the screen and sends sit to the Text-to-Speech module which gives vocal feedback to the user.

Each of the modules in the system can be sensitive to the global state of the dialogue (e.g. the GDN in focus, the list of active GDNs) through dynamic selection of its resources (e.g. using appropriate GDN dependent grammars). The information about the dialogue state can be obtained by reading the information published by the interaction manager.

POMDP Toolkit

Toolkit for the development of dialogue systems using POMDPs including a parser for specification files, an interactive simulator, and a performance simulator. Was developed at the Human Media Interaction research group of the University of Twente by Trung H. Bui, Dennis Hofs and Boris van Schooten.

Steps to take from specification to simulation:

- Specify the problem in a dialogue POMDP specification. Examples are included with this toolkit.
- Parse the specification file to generate a canonical POMDP file.
- Run a POMDP solver.
- Use the POMDP specification file and solution file in the simulator.

Methodology to Create Applications

This framework addresses not only the problem of creating the infrastructure for developing Multimodal Interfaces but also provides a methodology on how to proceed to create applications with that framework. The mGDNs and Rapid Prototyping Methodology are a very important contribution of this Framework. Worth mention is the inclusion on this framework of very recent evolutions on Dialog Managers (POMPD) [75, 76] and of Emotion recognition and handling [77].

Comparison

Comparison of several frameworks was condensed in tabular format in [73] and is reproduced in Table 7.

	ICARE – OI [7]	OpenInterface [2]	IMBuilder/ MEngine [5]	Flippo et al. [8]	Krahnstoever [14]	Quickset [8]	Phidgets [8]	Papier-Mâché [8]	Java Swing MM Extension	Service Counter System	HephaisTK
Architecture traits											
Finite state machine			x							x	x
Components	x	x					x		x		
Software agents				x		x					x
Fusion by frames					x						x
Symbolic-statistical fusion						x					
Programming mechanisms											
Programming via "hard coding"					x	x				x	
Programming via API				x			x	x	x		
Programming via configuration file											x
Visual Programming tool	x	x	x								
Characteristics											
Extensibility		x	x	x					x	x	x
Pluggability							x		x		x
Reusable components	x	x									x
Open Source	x	x						x		x	x

Table 7 – Comparison of different multimodal toolkits and architectures, from[73].

Languages

Dumas et al. [78] explores one of the possible ways to address the problem of how to best represent and model multimodal human-machine interaction: description languages, and some of their characteristics. In particular, the article tries to answer two questions:

- What would be the uses of description languages for multimodal interaction?
- How should such languages be able to describe best multimodal interaction and its distinctive features?

Answering this last question leads us to introduce a set of nine guidelines, covering different user- and system centered aspects that should be handled by such description languages. Their table, summarizing state of the art languages and their features, is reproduced in the next figure.

	Layers	Events	Time	Plasticity	Web-oriented	Error handling	Data modeling
EMMA							X
XISL		X	X		X		
ICO		X	X			X	X
UsiXML	X	X		X	X		
TeresaXML	X	X					
MIML	X				X		
NiMMiT	X	X	X				

Figure 6 - State of the art languages and their features (Dumas, Lalanne et al. 2010)

A number of the approaches revolve around the concept of a “multi-modal web”, enforced by the World Wide Web Consortium (W3C) Multimodal Interaction Activity and its proposed multimodal architecture. This theoretical framework describes major components involved in multimodal interaction, as well as potential or existent markup languages used to relate those different components. Many elements described in this framework are of practical interest for multimodal HCI practitioners, such as the W3C EMMA markup language, or modality-focused languages such as VoiceXML or InkML [78]. (Dumas, Lalanne et al. 2010)

The works of the W3C inspired the **XISL** XML language. XISL focuses on synchronization of multimodal input and output, as well as dialog flow and transition. XISL is a language targeted at web interaction, and offering a SMIL-like language for multimodal interaction; thus, it provides control over time synchronicity (e.g. with parallel or sequential playing), at least on the output side [78].

Another approach of the problem is the one of MIML (Multimodal Interaction Markup Language). One of the key characteristics of this language is its three-layered description of interaction, focusing on interaction, tasks and platform [78]. UsiXML follows a transformational approach for developing multimodal web user interfaces, also in the steps of the W3C. Four steps are achieved to go from a generic model to the final user interface [78]. This transformational approach is also used in Teresa XML [78].

At a higher level of modelling, NiMMiT is a graphical notation associated to a language used for expressing and evaluating multimodal user interaction [78].

Input Related

EMMA

Building multimodal interfaces remains a complex and highly specialized task. Typically these systems involve a variety of different input and output processing components, such as speech, gesture recognition, and dialog management among others. Communication among components is not standardized and makes it difficult (or impossible) to plug-and-play components from different vendors or research sites, making component building harder for authors and delaying the development of multimodal systems [79]. (Johnston 2009)

The W3C EMMA addresses this problem by providing a standardized XML representation language for encapsulating and annotating inputs to spoken and multimodal interactive systems. As such, it targets primarily data transfer between entities of a given multimodal system [78].

EMMA targets primarily data transfer between entities of a given multimodal system; in this regard, EMMA perfectly addresses input and output data source representation; in fact, this is the only language to fully address this [78].

Advantages:

EMMA is an XML markup language which provides mechanisms for capturing and annotating the various stages of processing of users' input [79].

One of its critical design features is that it does not attempt to standardize the semantic representation assigned to inputs, rather it provides a series of standardized containers for mode and application specific markup, and a set of standardized annotations for common metadata [79].

Its documents are not supposed to be directly created by authors; rather they are generated automatically by system components such as a multimodal fusion engine.

EMMA Tags:

On EMMA, there are two key aspects to the language: a series of elements (e.g. emma:group, emma:one-of, emma-interpretation) which are used as containers for interpretations of the users' inputs, and a series of annotation attributes and elements which are used to provide

various pieces of metadata associated with those inputs, such as timestamps (emma:start, emma:end) and confidence score values (emma:confidence) [79].

```
<emma:emma>
  <emma:interpretation
    id="int1"
    emma:medium="acoustic" emma:mode="voice"
    emma:function="dialog" emma:verbal="true"
    emma:start="1241035986246"
    emma:end="1241035989306"
    emma:confidence="0.8" emma:lang="en-US"
    emma:process="smm:type=asr&version=asr_eng2.4"
    emma:media-type="audio/amr; rate=8000">
     <emma:literal>comedy movies directed by woody
     allen and starring diane keaton</emma:literal>
  </emma:interpretation>
</emma:emma>
```

Figure 10 - Example of a EMMA document generated by a Multimodal Fusion Server (from [79]).

Figure 10 represents a (simplified, some attributes were removed) EMMA document, produced by an ASR server (from a speech application), containing a single recognition result.

On the basis of every EMMA document is a emma:emma tag which must contain either one or more emma:interpretation (representing each a given input) tags or an interpretation container such as emma:one-of (which indicates only one of the interpretations may be used), emma:sequence (which indicates that the included interpretations possess a temporal/chonological order) or emma:group (which indicates that the various contained interpretations are grouped by a given criteria) tag.

Plus, in each of the interpretations (or the container) exists a number of attributes that define the input. Examples of such attributes are emma:medium and emma:mode which classify the user's input modality (in the example, the medium is acoustic and the modality is voice). Emma:function which differentiates interactive dialog (dialog) from other uses such as recording and verification.

The attributes emma:start and emma:end indicate the start and end of the user's input signal in milliseconds. The emma:confidence tag represents the processor's confidence on the interpretation (between 0 and 1), that is, the quality of the input. The tag emma:literal is used to contain any given string on the EMMA document. And another element (not in the

example) worth mentioning is emma:tokens, which indicates the particular string of words that were recognized.

Figure 11 represents another EMMA document, this time generated by the multimodal fusion server based on the previous document (Figure 1). Here we can see the capabilities of EMMA for representing the relationships between multiple stages of processing. The element emma:derived-from provides a reference to the resource emma:interpretation from which this new emma:interpretation was derived. The element emma:derivation is used as a container for the earlier stage of processing. Note that any annotations which appear on the earlier stage of processing (int1) are assumed to apply to the later stage (int2) unless they are explicitly restated (such as emma:process and emma:confidence). Finally, the result of processing is (in this case) shown by the <query> tag which will be later used (by the client) [79].

```
<emma:emma>
  <emma:interpretation
    id="int2"
    emma:tokens="comedy movies directed by woody
                 allen and starring diane keaton"
    emma:confidence="0.7"
    emma:process="smm:type=fusion&version=mmfst1.0">
    <query><genre>comedy</genre>
           <dir>woody_allen</dir>
           <cast>diane_keaton</cast></query>
    <emma:derived-from resource="#int1"/>
  </emma:interpretation>
  <emma:derivation>
   <emma:interpretation id="int1"
     emma:medium="acoustic" emma:mode="voice"
     emma:function="dialog" emma:verbal="true"
     emma:start="1241035986246"
     emma:end="1241035989306"
     emma:confidence="0.8" emma:lang="en-US"
     emma:process="smm:type=asr&version=asr_eng2.4"
     emma:media-type="audio/amr; rate=8000">
      <emma:literal>comedy movies directed by woody
      allen and starring diane keaton</emma:literal>
   </emma:derivation>
  </emma:derivation>
</emma:emma>
```

Figure 11 - Example of a EMMA document generated by a Multimodal Fusion Server (from [79])

Modality Related

InkML

The Ink Markup Language [80] serves as the data format for representing ink entered with an electronic pen or stylus. The markup allows for the input and processing of handwriting, gestures, sketches, music and other notational languages in applications. It provides a common format for the exchange of ink data between components such as handwriting and gesture recognizers, signature verifiers, and other ink-aware modules.

VoiceXML

VoiceXML 3.0 [81], a modular XML language for creating interactive media dialogs that feature synthesized speech, recognition of spoken and DTMF key input, telephony, mixed initiative conversations, and recording and presentation of a variety of media formats including digitized audio, and digitized video.

EmotionML

EmotionML [82] will provide representations of emotions and related states for technological applications. As the web is becoming ubiquitous, interactive, and multimodal, technology needs to deal increasingly with human factors, including emotions. The language is conceived as a "plug-in" language suitable for use in three different areas: (1) manual annotation of data; (2) automatic recognition of emotion-related states from user behavior; and (3) generation of emotion-related system behavior.

Interaction Related

SMUIML

SMUIML (Synchronized Multimodal User Interaction Modelling Language), a description language for multimodal human-machine interaction... [78](Dumas, Lalanne et al. 2010)

This language has been created as a means for the developers wishing to use HephaisTK to easily access the deeper functionalities of the toolkit without having to delve into the code.

A typical SMUIML declares recognizers, triggers and actions, and the user-machine dialog in the form of a finite state machine calling those triggers and actions.

CARE properties are fully integrated into SMUIML and can be used to specify the way modalities will have to be fused, for example in a parallel or complementary way.

SMUIML stands for Synchronized Multimodal User Interaction Modelling Language. As its name implies, the language seeks to offer developers a language for describing multimodal interaction, expressing in an easy-to-read and expressive way the modalities used, the recognizers attached to a given modality, the user-machine dialog modelling, the various events associated to this dialog, and the way those different events can be temporally synchronized [78]. (Dumas, Lalanne et al. 2010)

SMUIML structure:

The way a SMUIML file is split allows a clear separation between three levels necessary to the integration process.

As shown below, < recognizers> are at the lower, input/output level, < triggers > and <actions> form a middle level, devoted to events management, and the upper level contains the <dialog> description.

```
<?xml version="1.0" encoding="UTF-8"?>
<smuiml>
<integration_description client="client_app">
<recognizers>
<!--...-->
</recognizers>
<triggers>
<!--...-->
</triggers>
<actions>
<!--...-->
</actions>
<dialog>
```

```
< ! −− . . . −− >

< / d i a l o g >

< / i n t e g r a t i o n _ d e s c r i p t i o n >

<smuiml>
```

This abstraction in three different levels allows components definition and reusability. In order to enhance reusability, the upper dialog level allows definition of clauses that can be later used and extended [78].

SMUIML recognizers:

At the recognizers' level, the goal is to tie the multimodal dialog scenario with the actual recognizers that the developer wishes to use for his application. In the context of the HephaisTK toolkit, all recognizers are identified by a general name throughout the toolkit. This general identifier is hence used in SMUIML. The HephaisTK toolkit keeps a list of recognizers, and their associated modality (or modalities) [78].

Triggers are at the core of the transition mechanism of SMUIML. They describe a sub-set of interest from all the possible events coming from the different recognizers. A set of input events can hence be abstracted behind one trigger name, enhancing as much the script readability [78].

SMUIML actions

<actions> are the output equivalent of < triggers >. They describe the messages and their content that will form the communication channel between HephaisTK toolkit and its client application.

SMUIML dialog

The <dialog> element describes the integration mechanisms of a SMUIML script. In essence, a <dialog> is a finite state machine, with transitions defined by the < triggers > and <actions> events that were presented in the former sections. States of the dialog are described by <context> elements. Each context has a unique name identifying it. One context must have a "start_context" attribute, defining it as the starting state. An "end_context" attribute also exists to describe a final state [78] (Dumas, Lalanne et al. 2010).

Output Related

SMIL

Synchronized Multimedia Integration Language (SMIL 1.0, pronounced "smile") [83] allows integrating a set of independent multimedia objects into a synchronized multimedia presentation. Using SMIL, an author can: describe the temporal behavior of the presentation; describe the layout of the presentation on a screen; associate hyperlinks with media objects.

MOXML - Multimodal Output eXtended Markup Language

In the MOSTe (Multimodal Output Specification Tool), the resulting specification is saved in a proprietary language for future use. This language called MOXML (Multimodal Output eXtended Markup Language), describes all the specification elements.

At the present time, the definition of an outputs specification is not managed by the W3C's Extended Multimodal Annotation Markup Language (EMMA). So MOSTe authors defined their own data representation language based on XML with a set of tags describing all needed elements in an output multimodal system.

A Representative State of the art project

The CALLAS project [84]

CALLAS (the acronym stands for Conveying Affectiveness in Leading-edge Living Adaptive Systems) is an integrated project funded by the European Commission under FP6 [85].

One of the main challenges for CALLAS is to implement the concept of affective emotional input for interactive media rather than within a traditional interface paradigm.

Affective and emotional interfaces are generally concerned with the real-time identification of user emotions to determine system response. They rely most often on Ekmanian emotions such as joy, fear or anger. However, interaction with new media such as interactive narratives, digital theatre or digital arts involves different ranges of emotions on the user's side, some of which correspond to responses to aesthetic properties of the media, or characterize the user experience itself in terms of enjoyment and entertainment. To identify these, more complex articulations of modalities are required across semantic dimensions as well as across temporal combinations.

Firstly, modalities involved range from emotional language and paralinguistic speech (laughter, cries) to categorizations of user attention (suggesting interest or boredom for instance).

Secondly, these have to be integrated across interaction sessions of variable durations rather than analyzing a single emotional status in real-time. One such example of integration consists

of affective input to interactive narrative, in which the evolution of a baseline plot can be influenced by user reactions to the story unfolding, analyzed in terms of overall attitudes (body postures, evolution of user activity, paralinguistic speech) [84].

CALLAS is based on a three-layer structure (Figure 12) that maps the general objectives into operational areas [86]:

- The **CALLAS Shelf**: a library of multimodal components developed and made available from the Consortium partners, starting from state-of-the-art technologies, improved and transformed into exploitable components.
- The **CALLAS Framework**: a plug-in architecture for the interoperability between the components, allowing multimodal applications developers to combine them at design time, providing significant cost reduction as well as quality of software improvement.
- The **CALLAS Showcase**: experimental applications using the CALLAS Framework to demonstrate how successful the technology is in conveying effectiveness and augmenting the people experience in different interactive spaces.

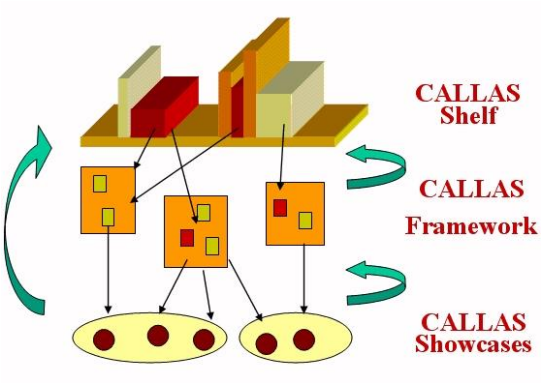

Figure 12 – CALLAS Architecture.

Sample State-of-the-Art Applications

With current technological advances, and the pervasiveness of cheaper ICTs, research in the area of multimodal applications has increased in the past few years, with main-stream support already available in some devices. Research applications range from, tabletops with multi-touch and voice interface support, allowing the development of a collaborative gaming environment [87], to multimodal media center interfaces [88] and AAL applications geared towards older and disabled users [11, 89].

More main-stream applications and devices, with support for multimodal interaction, include several iPhone accessibility oriented applications, which support regular interaction through a touch screen and voice recognition [90] and some Android applications such as Google Maps and Google Earth [91].

Archivus

Archivus [92] is a multimodal (voice, keyboard, mouse/pen) meeting browser, whose purpose is to allow users to access multimedia meeting data in a way that is most natural to them. Its

user interface design is based on the metaphor of a person interacting in an archive or library (Figure 13).

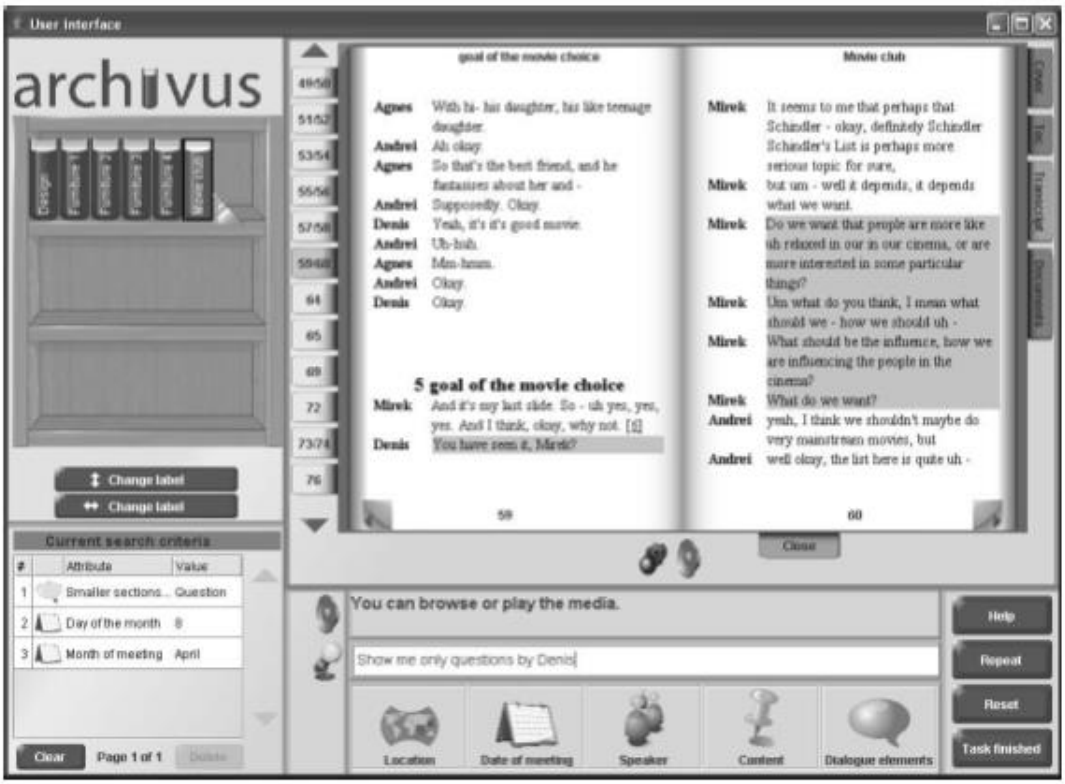

Figure 13 - User interface of the Archivus Browser

Archivus was implemented within a software framework for designing multimodal applications with mixed-initiative dialogue models Systems designed within this framework handle interaction with the user through a multimodal dialogue manager. The dialogue manager receives user input from all modalities (speech, typing and pointing) and provides multimodal responses in the form of graphical, textual and vocal feedback.

Its functioning is based on **two wizards**, an Input Wizard and an Output Wizard. The role of the Input Wizard is to assure that the user's input (in any modality combination) is correctly conveyed to system in the form of sets of semantic pairs. A semantic pair is a piece of information that the dialogue system is able to understand. For example, a user may ask "What questions did this guy ask in the meeting yesterday?" and at the same time point to a person on the screen called "Raymond". This could be translated to dialogact:Question, speaker:Raymond and day:Monday.

The Output Wizard's function is to monitor, and if necessary change the default prompts that generated by the system, for instances, in order to improve the dialogue flow and thus better explain the dialog situation to the user. For example if “Monday” is not a meeting date in the database, the input is interpreted as having “no match”, which generates the system prompt “I don’t understand”. Here, the Output Wizard can intervene by replacing that prompt by one that more precisely specifies the nature of the problem.

MOBILE (PDAs, Smartphones)

Over the past 10 years, the phone has expanded from being just a phone to being a full multimedia unit, on which you can play games, shoot photos, listen to music, watch television or video, send messages, and do video conferencing.

Most mobile devices currently support key-based interfaces through joypad and direction keys and a numerical keyboard. On larger devices, additional keys provide a better user experience for complex tasks because keys can be dedicated to specific tasks. Smart phones cannot easily use such keys owing to limited physical space. As such, interaction with touch sensitive screens has emerged as an alternative, leading into a multimodal application trend.

This section shows some examples of multimodal applications on mobile phones.

Mobile Browsing - Openstream's Cue-me

Built on open standards, Openstream's Cue-me browser enables multimodal mobile application development for various handsets such as Windows Mobile, Symbian and BlackBerry platforms. Cue-me provides an alternative to the small-keypad mode of interaction.

Cue-me provides a way to combine speech and gesture to make sure mobile users are able to convey intent quickly and easily.

With Cue-me, its possible to gesture at something and give a speech command so that the browser can understand what the user wants. With it, the browser can make the user's command happen without the user having to use the keypad.

A utilization of Cue-me is Openstream's **Clinical Trials solution**. Its goal is to improve the efficiency of clinical trials by enabling effective collaboration with subjects using mobile devices, exchange of rich data in real time, improving data accuracy and reducing rework and improving safety and compliance.

It allows field-personnel use multiple modes of input such as voice, camera images, video and digital ink annotations to promote richer communication and real-time collaboration among the players in the eco-system. Some key features are:

- Rich information using multi-modal collaboration for eDiary, Protocol forms and other requirements
- Increased Coverage by running applications on any device
- Automated data collection using Medical Device Integration

Multimodal biometric authentication on a phone

A Multimodal user authentication system implemented on a PDA [93] is part of the SecurePhone project.

The aim of the project is twofold. The first aim is to enable the secure exchange of written and spoken documents. By using private and public keys, a PDA user can send a document securely to another PDA user, who can then edit the document and send it back for further editing, until a final form of the document has been agreed. The second aim is to use biometric authorisation (rather than PIN) to confirm that the user is the registered owner before electronically signing the document.

The system relies on three modalities: **voice, face and signature**. These modalities were chosen because they are easy to acquire on a standard PDA and are all characterized by a high user acceptance. All preprocessing of the signals is performed on the PDA, while storage and processing of the clients biometric profile will all be done on the SIM-card in the PDA. Data on the SIM-card is accessible only to the service provider, so in this way the security of the biometric authentication is maximized. However, given the storage and processing limitations of presently available SIM-cards, strong restrictions are placed on the biometric authentication methods which can be used.

Mobile Gaming

A recent study [94] shows that smartphone owners are much more likely to be interested in gaming, not only playing more often but taking more interest in game genres as a whole. About 47.1% of smartphone owners play at least one game per month versus 15.7% on feature phones; 13.3 percent of those smartphone owners play every day. Most of the gaming habits skewed towards puzzles and traditional board games, but the usage rates across smartphone users was in all cases multiple times larger.

With the introduction of **touch and accelerometers** on mobile devices, games have become one of the most successful application types. For example, Figure 14 shows Aqua Forest, a unique application utilizing "Phyzios Engine", a 2D-based multi-physics engine for casual games. This engine uses a particle-based physics model that has few restrictions. It can calculate almost any type of objects, not only solid materials, but also elastic body, plastic body, fluid, and gas. In the "FREE" mode of the application, its possible to draw and create various shapes of objects and test the ability of game's engine. In the "PUZZLE" mode, with exploiting the touch screen and accelerometer, the challenge consists in completing the

puzzles designed by "Phyzios Engine". There are 5 categories, and each category has 10 puzzles.

Figure 14 - Aqua Forest - an iPhone game that supports touch and accelerometer based movements.

Siri - Mobile Personal Assistant

Siri (www.siri.com, [95]) brings a conversational interface to the iPhone which allows you to ask it to perform tasks for you such as find a French restaurant nearby and book a table, look up movie listings, order a taxi, or look up the phone number and address of a local business.

The user can simply speak into the phone with a request like, “Find something to do in San Francisco this weekend”. It turns the speech to text and pushes the request out to an appropriate service on the Web such as Eventful or Citysearch, in this case. It not only attempts to bring back the appropriate information based on context, time of day, and your location, but with the user's permission, it can go ahead and make reservations or buy tickets as well.

Siri combines an impressive array of technologies and brings them together on the iPhone. These include natural language processing and semantic analysis. In a way, Siri is the “mother of all mashups”. The iPhone app is a conversational interface with Siri’s servers on the Web, which tie into nearly 30 different APIs at launch, with more on the way. These include OpenTable, TaxiMagic, MovieTikets.com, Rotten Tomatoes, WeatherBug, Yahoo Local, Yahoo Boss, StubHub, Bing, Eventful Freebase, Citysearch, AllMenus.com, Gayot, and Wolfram Alpha.

Figure 15 - Siri User Interface.

Siri is a free application, centering its business model on the affiliate fees it receives every time the user buys something like a concert ticket or make a restaurant reservation through the app. In addition to helping the user do things, it also can be used to set reminders. The user can simply tell it to remind him by email to make a phone call on Thursday morning, and it can figure it out. The app licenses its speech-to-text engine from Nuance, another SRI spin-off.

Siri was born out of SRI's CALO Project, the largest Artificial Intelligence project in U.S. history. (CALO stands for Cognitive Assistant that Learns and Organizes). Made possible by a $150 million DARPA (Defense Advanced Research Projects Agency) investment, the CALO Project included 25 research organizations and institutions and spanned 5 years. Siri is bringing the

benefits of this technology to the public in the first mainstream consumer application of a virtual personal assistant.

Siri, Inc. was founded in 2007 and is based in San Jose, California. Siri is venture-backed by investors including Menlo Ventures, Morgenthaler Ventures, The Li Ka Shing Foundation, and SRI International.

Automotive

In recent years, the complexity of on-board and accessory devices, infotainment services, and driver assistance systems in cars has experienced an enormous increase. This development emphasizes the need for new concepts for advanced human-machine interfaces that support the seamless, intuitive and efficient use of this large variety of devices and services [2].

A modern car already implements hundreds of functions that a user can interact with, in some cases deployed over almost a hundred embedded platforms [2].

In the coming years speech recognition will be a commodity feature in car. Control of communication systems integrated in the car infotainment system including telephony, audio devices and destination inputs for navigation can be done via voice. Concerning speech recognition technology biggest the challenge is the recognition of large vocabularies in noisy environments using cost sensitive hardware platforms. Further intuitive dialog design coupled with natural sounding text to speech systems has to be provided to achieve a smooth man-machine interaction [96].

Siemens Speech Processing

Speech signals carry more than just words and sentences: there is implicit information about the speaker' gender, age, language, and mood or stress - which is of value for many applications. In order to make this information accessible, Siemens Speech developed components for speaker recognition and speaker characterization. While speaker recognition has to be trained on the person to be recognized (enrollment) speaker characterization derives age/gender or language decisions speaker independently.

With the event of cellular phones, processing power became cheap enough to bring speech recognition on mobile devices. For that purpose a dedicated recognizer product was developed that offers various benefits. The Siemens Recognizer Embedded is targeted for mobile phones, car infotainment and navigation, PDA/PNA deployment, and dedicated embedded systems in hearing aids, medical devices, or industrial panels and comes with selected European, US and Asian languages (see Figure 16).

Figure 16 - User interface for voice driven applications in car.

This Siemens Recognizer Embedded is complemented by the Siemens Recognizer Server that offers standard interfaces and protocols like MRCP and RTP, multi-port and multi-threading with load-balancing, and optimizations for Windows and Linux. The Siemens Recognizer Server is targeted for call-center automation, auto-attendant solutions, and industrial applications.

Nuance's Systems

High performance recognition opens new opportunities for a more natural interaction by voice in cars. When vocabularies are no longer restricted to few commands or names but extend to several thousand words, and when those recognizers are combined with an appropriate dialog engine and Text-to-Speech synthesizer, especially in the automotive scenario new speech applications become reality that will significantly enhance usability.

A typical use-case for speech recognition is the control of entertainment sources of car infotainment systems. Available radios already display the name of the tuned station, provided as the “Program Service Name” by the radio data system RDS.

The development of effective compression techniques for audio like MP3-coding and the availability of portable players, even integrate in various recent cell phones accelerated the demand to consume audio and video media wherever they are.

The use of speech control for the administration of large amounts of audio files, playback control, and the selection of titles and artist becomes a desirable feature, especially for the case of limited interaction possibilities of portable players or car infotainment systems.

A success example is Nuance's. Nuance is the world's #1 supplier of multimodal input and output solutions for automotive and navigation systems. Their solutions, which span speech recognition, text-to-speech, signal enhancement, predictive text and more, provide state-of-the-art interfaces to in-car navigation, entertainment and telematics systems to keep drivers safe behind the wheel. Nuance's automotive speech solutions have been successfully implemented in more than 5 million cars worldwide, representing more than 100 automobile models from more than 25 automobile brands from all major car manufacturers, including DaimlerChrysler, Fiat, Ford, Nissan and Renault, as well as quality Tier 1 suppliers, such as Aisin AW, Alpine, Bosch Blaupunkt, Bury, Denso, Magneti Marelli and Microsoft.

Figure 17 - Audi A8 MMI User Interface.

For example, Nuance's innovative speech technology have recently been integrated into the Audi MultiMedia Interface (MMI) Touch in-car infotainment system for navigation, media and phone as a way to provide an easy-to-use interface that minimizes visual and manual distractions behind the wheel. MMI consists of a single interface, which controls a variety of devices and functions of the car, thus minimizing the vast array of buttons and dials normally found on a dashboard. The system consists of the MMI terminal and the MMI display screen.

Some of its speech-enabled features in the new 2010 Audi A8 (Figure 17) models include:

One-Shot Destination Entry: With Nuance One-Shot Destination Entry, drivers can enter an entire destination address in one, simple spoken command. For example, just say "London, Downing Street, 10" and the navigation system in the MMI Touch will begin the route. Systems deployed in the U.S. will allow drivers to say "street in vicinity", eliminating the need to even input city and state.

Drivers will also have access to their address book. Simply say "Navigate to John Smith, home address", and the system will begin the route.

Music Search: Nuance Music Search enables a safer and more enjoyable interaction with the Audi A8's infotainment system by giving drivers the ability to access their favourite, stored songs by speaking the audio source, genre, artist, album or song with one simple, spoken command. For instance, just say "Play artist Lady Gaga" or "Play title Bad Romance". Nuance Music Search features multilingual speech recognition to respond to several languages in parallel. Drivers are also able to set radio stations by name or frequency, and play the CD, DVD and MP3 players with simple voice commands.

Address book and phone: Nuance's speech capabilities also enable voice-dialing. Drivers can store upwards of 2000 contact entries and assign up to 50 individual name tags to make selecting the most commonly accessed contacts even easier by voice, like "Mom", or "Work" – it's completely customizable.

Assistive Living

To finish this section we present the application area more closely related to the Living Usability Lab project, Ambient Assistive Living (AAL).

i2home - i2home.org

The i2home project is an approach based on existing and evolving industry standards. It focuses on the use of home appliances and consumer electronics by persons with cognitive disabilities and older persons [97].

Most of the functionalities offered by modern automation solutions, that in fact are intended to ease the everyday use of home appliances, are too complicated to be communicated understandably to the end users. This situation is particularly disadvantageous for elderly and disabled persons, namely the group of users that should benefit the most from modern technologies.

A key concept in i2home is the **"Pluggable User Interface**" approach that makes it easy to have a uniform design of the user interface and at the same time to have control over different home devices and appliances. The design where the user interface is separated from backend services and devices makes adaption and substitution of user interfaces and their components possible. To this end, the user interface can be exchanged, attached or detached at runtime

as appropriate, e.g., a user operates a PDA by click gestures while another user prefers to interact via voice control with the same device.

The i2home system architecture is based on the Universal Control Hub (UCH) which implements the URC framework in the home environment. The UCH represents the control center in i2home enabling the communication between any devices for interacting with the digital home and any backend devices that should be manipulated and/or monitored. A resource server makes pluggable user interfaces available that can be downloaded in order to apply the favored user interface by request. Currently, the i2home system includes target devices such as the TV, **HVAC (Heating, Ventilation, Air Conditioning)** the hood, a blood sugar meter, a calendar and an EPG service that are monitored by a single controller, a smartphone device.

The smartphone represents a generic controller equipped with a large screen that supports e.g. the customization of the button sizes so as to satisfy the preferences of i2home's target users. Additionally, an embedded calendar and reminder functionality supports the user in everyday life. The user can add new calendar entries, e.g. for taking pills, brush the teeth or visit the doctor. The built in alarm function reminds him to his keep appointments regardless which menu is currently displayed.

Figure 18 - The i2home user interface for multimodal interaction on a smartphone.

The user interface (Figure 18) jointly developed by the German Research Center for Artificial Intelligence (DFKI GmbH) and the Swedish Institute of Assistive Technology (SIAT) consists of a multimodal user interface implemented on a HTC Advantage smartphone. It allows interactions as combinations of click gestures and speech.

The system takes the role of a mediator between user and application. If the user wants to switch the TV channel, e.g. to CNN, the simple commando “Switch to CNN” suffices to switch the channel, independent from the active graphical menu. Furthermore the actual context is regarded when interpreting speech input. If the display shows the graphical menu for the air condition the command “Turn o” activates the air condition and not the TV or any other appliance.

A Multimodal Pervasive Framework for Ambient Assisted Living

The framework presented in [98] framework proposes a configurable, scalable, adaptive and multimodal framework, based on a grammar-based paradigm, which enables the user to have a more natural interaction with the system in the context of pervasive applications. The framework is applied in the field of Ambient Assisted Living (AAL) in order to provide a personal assistance for independent living and active ageing of cognitive impaired people.

The framework supports voluntary and involuntary multimodal interaction. The voluntary multimodal interaction consists of the explicit I/O process between the aged people and the multimodal system. The involuntary multimodal interaction involves all information that aged people and environment implicitly exchange with the multimodal system.

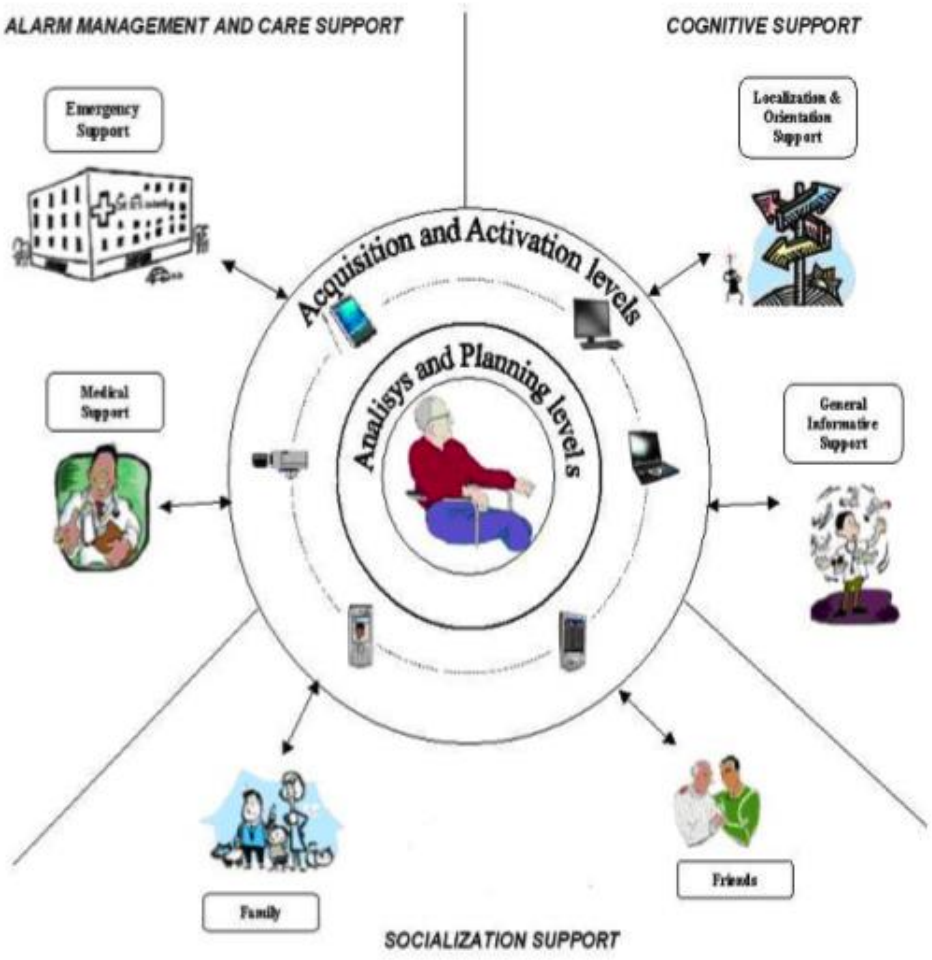

Figure 19 . AAL Framework Operational Scenario

The framework provides the following kinds of **support to elderly people**:

Cognitive Support: provides a user with physical disability the possibility to express multimodal query in order to have a general informative support. For example, if the user needs to have information about TV programs, he/she can obtain it by expressing the vocal command “I want to know the TV programs of this evening”.

The framework also enables the user to obtain localization and orientation support in unknown outdoor and indoor environments. The goal is to facilitate elderly people orientation in everyday life.

Socialization Support: allows the user both to communicate with relatives and friends by using different modalities and devices. For example, if the user needs to perform voice dialling and other phonebook control functions, such as to save or update telephone numbers, he/she can interact with the framework by the synchronized use of speech, handwriting, and/or pointing gesture modalities.

Care Support: supports the user in order to avoid mistakes in the administration of prescribed medication along the day. For example, if the user needs information about the dosage and time of a specific drug administration, he/she can interact with the framework by speech and/or pointing gesture modalities. Therefore, the user might express the vocal command "At what time must I take this?" while pointing the drug icon on a touch-screen display.

In order to test the effectiveness of the framework, a **usability evaluation has been performed**. The results of this evaluation proved that the use of the framework has a positive impact on the majority of users, and it has beneficial effects in terms of naturalness of interaction and quality of life of cognitive impaired people.

Conclusions

Main Research problems / Challenges

Despite the availability of multimodal devices, there are still very few commercial multimodal applications. One major reason for this is perhaps the lack of a framework that helps to create and develop multimodal applications in reasonable time and with limited resources.

In fact, up until now, most tools on this area were created for specific interaction paradigms, such as, camera-based interaction or tangible interfaces [99].

The most important investigation problems in this area with relevance for the Living Usability Lab project are:

1. How to use multimodal interaction for universal access effectively?
   Speech focused Multimodal Interaction investigation in the FP5 COMIC project [100] has shown that the universal access problem is much more complex than adding modalities.

2. How to rapidly develop applications making use of multimodal interaction?
   The answer to this question resides on building simple and general use tools.

3. How to develop universal forms for interfaces definitions?

Delegating for "renderers" the process of creating the final shape of the interface, appropriately adapted to the capabilities of the devices and methods available in each moment.

4. How to perform the management of multimodal Interaction/Dialogue?
   If the creation of dialogue managers for interfaces based on speech is a big challenge, the generalization to various modalities is - and will remain for many years - a bigger one.

5. How to have a bigger input/output balancing?
   In most uses of multimodal, there is a preponderance of input modalities.

6. Development of modalities capable of "handsfree" operations, permitted by advances in speech technologies (recognition and synthesis)
   As a modality, speech is very different from graphical interfaces. The biggest difference is navigation on the interface. In visual interfaces, much more information and navigation options may be presented simultaneously. Moreover, voice/speech commands allow shortcuts much more intuitively. Despite voice interfaces being promising, without due care and without a use according to their specificities generally results in interfaces with poor user acceptance. It is necessary to invest in the creation of tools, methods and best practices to make real the promises of this modality.

7. How to perform fusion of multiple modalities and integrate inputs and outputs
   With recent developments in multimodal interfaces, various approaches have been proposed to fuse the input data and for generating output. However, less attention was devoted on how to integrate them into an input and output multimodal system. [101] propose an approach, called THE HINGE, allowing outputs which take into account the results of merging the entries.

8. Creating solutions for a larger set of (Human), including Portuguese
   Much of what has been done in these areas is not liable for direct use in all languages, making it necessary, for example, to adapt some of the modalities such as speech, to other languages.

9. Improve unimodal
   The state-of-the-art continuous speech recognition and gesture recognition are still very far from the human's abilities. They can work quite reliable in "ideal" laboratory conditions, but then one tries to apply them for real exploitation they essentially lose in quality. Key challenges for speech recognition are robust voice activity detection, noise suppression and speaker localization amongst others. Problems for gesture recognition are, for instances, change of illumination conditions, dynamic background or presence of several persons in the image. All these topics must be comprehensively studied in order to improve the quality of speech and gesture interfaces [102].

10. Error Handling
    Error handling currently remains one of the main interface problems for recognition-based technologies. However if there is some redundancy in inputs, it is possible to apply methods of mutual disambiguation between the signals. Mutual disambiguation involves recovery from unimodal recognition errors within a multimodal architecture, because semantic information from each input mode supplies partial disambiguation of the other mode, thereby leading to more stable and robust overall system performance [102].

11. Adaptation
    The diversity of environments, systems and user profiles leads to a contextualization of the interaction. Initially the interaction had to be adapted to a given application and for a specific interaction context. Nowadays, the interaction has to be adapted to different situations and to a context in constant evolution [70].

    This diversity of the interaction context emphasizes the complexity of a multimodal system design. It requires the adaptation of the design process and more precisely the implementation of a new generation of user interface tools. These tools should help the designer and the system to make choices on the interaction techniques to use in a given context [70].

    Efficient multimodal interfaces should be able to take into account user's requirements and needs. Fast automatic adaptation to user's voice parameters, height, skin color, clothes is a very important of prospective speech and gestures multimodal systems. An ability of an interface to recognize current context, to change dialogue modal in correspondence with this information, as well as to process out-of-vocabulary voice commands, semantically rich gestures and to add dynamically new items in the recognition vocabulary should be the object of the further studies too [102].

12. Inclusion of additional modalities
    Other natural modalities could help increasing the accuracy robustness of the interfaces involving modalities such as gesture or speech. For instance, eye gaze can be considered as a complementary pointer to a spatial object on the screen. Moreover usage of facial expressions (for instance, lip motions) could enhance the automatic speech recognition especially in acoustical noisy environments. An interface able to process both modalities in parallel is known as audio-visual speech recognition [102].

13. Creation of Methods for evaluation and usability of the interfaces
    Some ISO 9241 standards are in the process of being elaborated and concern instance testes for evaluation command dialogues, direct manipulation dialogues, haptic and

tactile interactions in multimodal environments. Additional researches are needed to create suitable tests for comparison between contactless multimodal interfaces.

On a more generic standpoint, design issues such as input and output selection, avoiding supplying contradictory or redundant data to the user, user and environment adaptability, consistency and error handling should be thought out carefully, especially when dealing with disabled users [50, 103].

These generic issues can be caused by several technical issues, ranging from input recognition errors, system delays, fusion engine issues, or even a combination of these factors [17].

Also, a well-designed multimodal system should be able to deal with imperfect or incomplete data, having the ability to infer conclusions from this data with some certainty. This effect, called multimodal disambiguation, can be done through probabilistic methods such as HMM's, Bayesian networks, or Dynamic Bayesian networks, which are capable of dealing with noisy information, temporal information as well as missing data, using probabilistic inference. More direct and simpler ways of dealing with ambiguity exist, ranging from asking the user, through another modality, what option better suits his previous input, always choosing the first available option or giving preference to one particular modality over another such as, speech over gestures, or vice-versa [13, 50].

REFERENCES

1. Sun, Y., et al., *An efficient unification-based multimodal language processor in multimodal input fusion*, in *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces*. 2007, ACM: Adelaide, Australia. p. 215-218.

2. Müller, C. and G. Friedland, *Multimodal interfaces for automotive applications (MIAA)*, in *Proceedings of the 13th international conference on Intelligent user interfaces*. 2009, ACM: Sanibel Island, Florida, USA. p. 493-494.

3. Bolt, R.A., *"Put-That-There";: Voice and gesture at the graphics interface*, in *Proceedings of the 7th annual conference on Computer graphics and interactive techniques*. 1980, ACM: Seattle, Washington, United States. p. 262-270.

4. Oviatt, S., *Ten myths of multimodal interaction.* Commun. ACM, 1999. **42**(11): p. 74-81.

5. Honkala, M. and M. Pohja, *Multimodal Interaction with XForms*, in *The Sixth International Conference on Web Engineering (ICWE2006)*. 2006: Palo Alto, California. p. 201-208.

6. Oviatt, S., *Multimodal interactive maps: designing for human performance.* Hum.-Comput. Interact., 1997. **12**(1): p. 93-129.

7. Oviatt, S., *Taming recognition errors with a multimodal interface.* Commun. ACM, 2000. **43**(9): p. 45–51.

8. Oviatt, S., *Designing robust multimodal systems for universal access*, in *WUAUC'01 - 2001 EC/NSF workshop on Universal accessibility of ubiquitous computing*. 2001. p. 71–74.

9. Oviatt, S., et al., *Designing the user interface for multimodal speech and gesture applications: State-of-the-art systems and research directions for 2000 and beyond.* Human-Computer Interaction in the New Millennium, 2000.

10. D'Andrea, A., et al. *A multimodal pervasive framework for ambient assisted living*. in *PETRA '09: Proceedings of the 2nd International Conference on PErvsive Technologies Related to Assistive Environments*. 2009. New York, NY, USA: ACM.

11. Salces, F.J.S.d., D. England, and D. Llewellyn-Jones, *Designing for all in the house.* , in *CLIHC '05: 2005 Latin American conference on Human-computer interaction,* . 2005. p. 283–288.

12. Bernsen, N.O., *Multimodality theory.* , in *Multimodal user interfaces. From signals to interaction*, D. Tzovaras, Editor. 2008, Springer. p. 5-29.

13. Lalanne, D., et al., *Fusion engines for multimodal input: a survey*, in *Proceedings of the 2009 international conference on Multimodal interfaces*. 2009, ACM: Cambridge, Massachusetts, USA. p. 153-160.

14. Serrano, M. and L. Nigay, *Temporal aspects of CARE-based multimodal fusion: from a fusion mechanism to composition components and WoZ components*, in *Proceedings of the 2009 international conference on Multimodal interfaces*. 2009, ACM: Cambridge, Massachusetts, USA. p. 177-184.

15. Stephanidis, C., et al., *Universal accessibility in HCI: Process-oriented design guidelines and tool requirements*, in *4th ERCIM Workshop on "User Interfaces for All"*. 1998: Stockholm, Sweden. p. 19-21.

16. Richter, K. and M. Hellenschmidt, *Interacting with the ambience: Multimodal interaction and ambient intelligence.* Interaction, 2004. **19**: p. 20.

17. Dumas, B., R. Ingold, and D. Lalanne, *Benchmarking fusion engines of multimodal interactive systems*, in *Proceedings of the 2009 international conference on Multimodal interfaces*. 2009, ACM: Cambridge, Massachusetts, USA. p. 169-176.

18. Bui, T.H., *Multimodal Dialogue Management - State of the art*. 2006, Human Media Interaction Department, University of Twente.

19. Ballagas, R., et al., *The smart phone: A ubiquitous input device.* Ieee Pervasive Computing, 2006. **5**(1): p. 70-77.

20. Minker, W., R. López-Cózar, and M. McTear, *The role of spoken language dialogue interaction in intelligent environments.* Journal of Ambient Intelligence and Smart Environments, 2009. **1**: p. 31–36.

21. Deketelaere, S., R. Cavalcante, and J.F. RasaminJanahary, *OASIS Speech-based interaction module*. 2009.

22. Bernsen, N.O., *Towards a tool for predicting speech functionality.* Speech Commun., 1997. **23**(3): p. 181-210.

23. Fabbrizio, G.D., T. Okken, and J.G. Wilpon, *A speech mashup framework for multimodal mobile services*, in *Proceedings of the 2009 international conference on Multimodal interfaces*. 2009, ACM: Cambridge, Massachusetts, USA. p. 71-78.

24. Militello, S. and S. Mosso, *Removing Barriers from Technology: Speech Technology and Disabilities*. 2006, Loquendo.

25. Teixeira, A., et al., *Speech as the Basic Interface for Assistive Technology* in *DSAI 2009 - Proceedings of the 2th International Conference on Software Development for Enhancing Accessibility and Fighting Info-Exclusion,* . 2009: Porto Salvo, Portugal.

26. Sun, *Java Speech API Programmer's Guide, Version 1.0* 1998. p. Chapter 3.

27. Hamill, M., et al., *Development of an automated speech recognition interface for personal emergency response systems.* Journal of NeuroEngineering and Rehabilitation 2009. **6**.

28. Microsoft. *Speech API*. 2008 [cited 2010 21 April 2010]; Available from: http://www.microsoft.com/speech/developers.aspx#none.

29. Microsoft, *Unified Communications Managed API.* 2008.

30. Pacifici, R., *Loquendo MRCP Server: the importance of being standard*, Loquendo.

31. Potter, R.L., L.J. Weldon, and B. Shneiderman., *Improving the accuracy of touch screens: an experimental evaluation of three strategies*, in *CHI '88: Proceedings of the*

SIGCHI conference on Human factors in computing systems. 1988, ACM: New York, NY, USA, . p. 27–32.

32. Hormby, T., *The story behind Apple's Newton*. 2006.

33. Royea, D., *Evolutionary tree*. 2005.

34. Hall, M. *Ce 6.0 - why the codename "yamazaki" ?* 2006; Available from: http://blogs.msdn.com/mikehall/archive/2006/09/19/763146.aspx.

35. FingerWorks, *igesture retro*. 2001.

36. Webster, S., *Droid does multitouch, milestone does it better*. 2009.

37. Microsoft. *Microsoft Surface*. 2009 [cited 2010 5 April]; Available from: http://www.microsoft.com/surface/en/us/Pages/Product/WhatIs.aspx.

38. Wang, F., et al., *Detecting and leveraging finger orientation for interaction with direct-touch surfaces*, in *Proceedings of the 22nd annual ACM symposium on User interface software and technology*. 2009, ACM: Victoria, BC, Canada. p. 23-32.

39. touchscreenmonitorguide.co.uk. *What Are The Benefits Of A Touch Screen Monitor For Business Use*. 2010 [cited 2010 April]; Available from: http://www.touchscreenmonitorguide.co.uk/benefits-touch-screen-monitor-business/.

40. Microsoft, *Microsoft Surface SDK 1.0 SP1 Workstation Edition*. 2009.

41. NextWindow. *NextWindow Two-Touch API*. 2008 [cited 2010 April]; Available from: http://www.nextwindow.com/support/application_notes/api.html.

42. Nesselrath, R. and J. Alexandersson, *A 3D Gesture Recognition System for Multimodal Dialog Systems*, in *Proceedings of the sixth IJCAI Workshop ``Knowledge and Reasoning in Practical Dialogue Systems* 2009: Pasadena, CA. p. 46-51.

43. Metcalf, J., *E3 2009 : Microsoft at E3 Several Metric Tons of Press Releaseapalloza*. 2009.

44. Hirsch, M., et al., *Bidi screen: a thin, depth-sensing lcd for 3d interaction using light fields. I*, in *SIGGRAPH Asia '09*. 2009. p. 1-9.

45. Saponas, T.S., et al., *Enabling always-available input with muscle-computer interfaces*, in *UIST '09 - 22nd annual ACM symposium on User interface software and technology*. 2009. p. 167–176.

46. Savov, V. *MIT gestural computing makes multitouch look old hat*. 2009; Available from: http://www.engadget.com/2009/12/11/mit-gestural-computing-makes-multitouch-look-old-hat/.

47. Westeyn, T., et al., *Georgia tech gesture toolkit: supporting experiments in gesture recognition*, in *Proceedings of the 5th international conference on Multimodal interfaces*. 2003, ACM: Vancouver, British Columbia, Canada. p. 85-92.

48. Poppinga, B. and T. Schlömer, *wiigee - A Java-based gesture recognition library for the Wii remote*. 2008: http://www.wiigee.org/index.html.

49. Schlömer, T., et al., *Gesture Recognition with a Wii Controller*, in *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction.* 2008: Bonn, Germany.

50. Jaimes, A. and N. Sebe, *Multimodal human-computer interaction: A survey.* Comput. Vis. Image Underst., 2007. **108**(1-2): p. 116-134.

51. Chan, M.T. *Automatic lip model extraction for constrained contour-based tracking*. in *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*. 1999.

52. Choraś, M., *Human Lips Recognition*. 2007. p. 838-843.

53. Yoshida, T. and S. Hangai, *Development of Infrared Lip Movement Sensor for Spoken Word Recognition.* Journal of Systemics, Cybernetics and Informatics, 2007.

54. Cowie, R., et al., *Emotion recognition in human-computer interaction.* Signal Processing Magazine, IEEE, 2002. **18**(1): p. 32-80.

55. Yang, Y., et al., *Facial expression recognition and tracking for intelligent human-robot interaction.* Intelligent Service Robotics, 2008. **1**(2): p. 143--157.

56. Ryan, A., et al., *Automated Facial Expression Recognition System.* IEEE International Carnahan Conference on Security Technology, 2009.

57. Valenti, R., A. Jaimes, and N. Sebe. *Facial expression recognition as a creative interface*. in *IUI '08: Proceedings of the 13th international conference on Intelligent user interfaces*. 2008. New York, NY, USA: ACM.

58. Dumas, B., D. Lalanne, and S.L. Oviatt, *Multimodal Interfaces: A Survey of Principles, Models and Frameworks*, in *Human Machine Interaction, Research Results of the MMI Program*, D.L. and and J. Kohlas, Editors. 2009, Springer. p. 3-26.

59. Traum, D. and S. Larsson, *The Information State Approach to Dialogue Management.*, in *Current and New Directions in Discourse & Dialogue* Smith and Kuppevelt, Editors. 2003, Kluwer Academic Publishers. p. 325-353.

60. Jurafsky, D. and J.H. Martin, *Dialog and Conversational Agents* in *Speech and Language Processing*, P. Hall, Editor. 2008.

61. Bohus, D. and A.I. Rudnicky, *The RavenClaw dialog management framework: Architecture and systems.* Comput. Speech Lang., 2009. **23**(3): p. 332-361.

62. Larsson, S. and D. Traum, *Information state and dialogue management in the TRINDI Dialogue Move Engine Toolkit.* Natural Language Engineering 2000(Special Issue on Best Practice in Spoken Language Dialogue Systems Engineering,): p. 323-340.

63. Bos, J., et al., *DIPPER: Description and Formalisation of an Information-State Update Dialogue System Architecture*, in *4th SIGdial Workshop on Discourse and Dialogue*. 2003. p. 115--124.

64. Williams, J.D. and S. Young, *Partially observable Markov decision processes for spoken dialog systems.* Computer Speech and Language, 2007. **21**: p. 393-422.

65. Chu, S.-W., et al., *An approach to Multi-Strategy Dialogue Management*, in *InterSpeech*. 2005. p. 865-868.

66. Ferguson, G., et al., *CARDIAC: An Intelligent Conversational Assistant for Chronic Heart Failure Patient Heath Monitoring.*, in *Proceedings of the AAAI Fall Symposium on Virtual Healthcare Interaction, .* 2009: Arlington, VA.

67. Lopes, L.S., et al. *INTEGRATED CAPABILITIES FOR KNOWLEDGE ACQUISITION THROUGH SPOKEN LANGUAGE INTERACTION IN A MOBILE ROBOT*. 2009.

68. Dumas, B., D. Lalanne, and R. Ingold, *HephaisTK: a toolkit for rapid prototyping of multimodal interfaces*, in *Proceedings of the 2009 international conference on Multimodal interfaces*. 2009, ACM: Cambridge, Massachusetts, USA. p. 231-232.

69. Foster, M.E., *State of the art review: Multimodal fission*. 2002.

70. Rousseau, C., et al., *Multimodal output specification / simulation platform*, in *Proceedings of the 7th international conference on Multimodal interfaces*. 2005, ACM: Torento, Italy. p. 84-91.

71. W3C, *Speech Synthesis Markup Language (SSML) Version 1.1*. 2010, W3C.

72. Lawson, J.-Y.L., et al., *An open source workbench for prototyping multimodal interactions based on off-the-shelf heterogeneous components*, in *Proceedings of the*

1st ACM SIGCHI symposium on Engineering interactive computing systems. 2009, ACM: Pittsburgh, PA, USA. p. 245-254.

73. Dumas, B., et al., *Strengths and weaknesses of software architectures for the rapid creation of tangible and multimodal interfaces*, in *Proceedings of the 2nd international conference on Tangible and embedded interaction*. 2008, ACM: Bonn, Germany. p. 47-54.

74. Bui, T.H., M. Rajman, and M. Melichar, *Rapid Dialogue Prototyping Methodology*, in *Proceedings of the 7th International Conference on Text, Speech and Dialogue*. 2004, Springer Verlag: Berlin. p. 579-586.

75. Bui, T.H., et al., *A POMDP approach to Affective Dialogue Modeling*, in *Fundamentals of Verbal and Nonverbal Communication and the Biometric Issue*, A. Esposito, et al., Editors. 2007, IOS Press: Amsterdam. p. 349-355.

76. Bui, T.H., et al., *A tractable hybrid DDN-POMDP approach to affective dialogue modeling for probabilistic frame-based dialogue systems.* Natural Language Engineering 2009. **15**(2): p. 273-307.

77. Bui, T.H., *Toward Affective Dialogue Management using Partially Observable Markov Decision Processes.* 2008, University of Twente.

78. Dumas, B., D. Lalanne, and R. Ingold, *Description languages for multimodal interaction: a set of guidelines and its illustration with SMUIML.* Journal on Multimodal User Interfaces, 2010. **3**(3): p. 237-247.

79. Johnston, M., *Building multimodal applications with EMMA*, in *Proceedings of the 2009 international conference on Multimodal interfaces*. 2009, ACM: Cambridge, Massachusetts, USA. p. 47-54.

80. W3C. *Ink Markup Language (InkML), W3C Working Draft*. 2006; Available from: http://www.w3.org/TR/InkML/.

81. W3C. *Voice Extensible Markup Language (VoiceXML) 3.0, W3C Working Draft*. 2008.

82. W3C. *Emotion Markup Language (EmotionML) 1.0, W3C Working Draft*. 2009.

83. W3C. *Synchronized Multimedia Integration Language (SMIL) 1.0 Specification* 1998.

84. Bertoncini, M. and M. Cavazza. *Emotional Multimodal Interfaces for Digital Media: the CALLAS challenge*. 2010 [cited 2010 2010/03/30]; Available from: http://www.callas-newmedia.eu/res/files/publications/callaschallenge.pdf.

85. CALLAS. *The CALLAS Project*. 2008 2008/September/18 [cited 2010 2010/April/18]; Available from: http://www.callas-newmedia.eu/about.html.

86. CALLAS. *Insight the CALLAS project*. 2008 2008/August/29 [cited 2010 2010/April/18]; Available from: http://www.callas-newmedia.eu/insights.html.

87. Tse, E., et al., *Exploring true multi-user multimodal interaction over a digital table*, in *DIS '08: 7th ACM conference on Designing interactive systems*. 2008. p. 109–118.

88. Turunen, M., et al., *Multimodal interaction with speech, gestures and haptic feedback in a media center application*, in *Human-Computer Interaction - INTERACT 2009*. 2009. p. 836–837.

89. Chotard, L. *Midas: Multimodal interfaces for disabled and ageing society tailoring solutions based on friendly, adaptive interfaces*. 2010; Available from: http://www.midas-project.com/.

90. Maccessibility.net. *Accessible Apps*. 2010; Available from: http://maccessibility.net/iphone/apps/.

91. Google. *Google Maps Navigation*. 2010; Available from: http://www.google.com/mobile/navigation/.

92. Lisowska, A., M. Rajman, and T.H. Bui, *ARCHIVUS: A System for Accessing the Content of Recorded Multimodal Meetings*, in *Proceedings of the 1st MLMI*. 2005, Springer. p. 291-304.

93. Koreman, J., et al., *Multi-modal biometric authentication on the SecurePhone PDA.* 2006.

94. Electronista. *Smartphones taking over in mobile gaming*. 2010; Available from: http://www.electronista.com/articles/10/04/14/study.shows.feature.phones.dying.in.games/.

95. Siri_Inc, *Siri - Mobile Personal Assistant*. 2009.

96. Tan, Z.-H. and B. Lindberg, *Automatic Speech Recognition on Mobile Devices and over Communication Networks (Advances in Pattern Recognition)*. 2008: Springer Publishing Company, Incorporated.

97. Nesselrath, R., et al., *Homogeneous Multimodal Access to the Digital Home for People with Cognitive Disabilities*, in *Ambient Assisted Living - AAL. 2.* . 2009: Berlin, Germany.

98. D'Andrea, A., et al., *A multimodal pervasive framework for ambient assisted living*, in *Proceedings of the 2nd International Conference on PErvsive Technologies Related to Assistive Environments*. 2009, ACM: Corfu, Greece. p. 1-8.

99. Serrano, M., et al., *The OpenInterface framework: a tool for multimodal interaction.* , in *CHI '08: CHI '08 extended abstracts on Human factors in computing systems*. 2008, ACM. p. 3501-3506

100. den Os, E.A. and L.W.J. Boves, *Natural multimodal interaction for design applications* in *Adoption and the Knowledge Economy* P. Cunningham, M. Cunningham, and P. Fatelnig, Editors. 2004, IOS Press: Amsterdam. p. 1403-1410

101. Sun, Y., et al., *THE HINGE between Input and Output: Understanding the Multimodal Input Fusion Results In an Agent-Based Multimodal Presentation System*, in *CHI 2008*. 2008.

102. Karpov, A., et al., *Two Similar Different Speech and Gestures Multimodal Interfaces*. 2009.

103. Reeves, L.M., et al., *Guidelines for multimodal user interface design.* Commun. ACM, 2004. **47**(1): p. 57-59.

104. Holzinger, A. (2003). Finger instead of mouse: touch screens as a means of enhancing universal access. In Proceedings of the User interfaces for all 7th international conference on Universal access: theoretical perspectives, practice, and experience, ERCIM'02, Berlin, Heidelberg, pp. 387–397. Springer-Verlag.

105. Murata, A. and Iwase, H. 2005. Usability of touch-panel interfaces for older adults. Hum Factors. 47, 4, 767—776.

106. Stone, R. (2008). Mobile touch interfaces for the elderly. Paper presented at the IADIS International Conference ICT, Society and Human Beings 2008, Amsterdam, The Netherlands.

107. Henze, N., Rukzio, E., and Boll, S. (2012). Observational and experimental investigation of typing behaviour using virtual keyboards for mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2659–2668, New York, NY, USA. ACM.

108. Werner, F., K. Werner, and J. Oberzaucher (2012). Tablets for seniors - an evaluation of a current model (ipad). In R. Wichert and B. Eberhardt (Eds.), Ambient Assisted Living, Advanced Technologies and Societal Change, pp. 177–184. Springer Berlin Heidelberg.

109. Loureiro, B. and R. Rodrigues (2011). Multi-touch as a natural user interface for elders: A survey. In Information Systems and Technologies (CISTI), 2011 6th Iberian Conference on, pp. 1–6.

110. Silveira, M. Técnica de Navegação em Documentos Utilizando Microsoft Kinect. 2011, Universidade Federal do Rio Grande do Sul.

111. Guerreiro, T. and Jorge, J., "Assistive technologies for spinal cord injured individuals: Electromyographic mobile accessibility," Proceedings of GW 2007, 7th International Workshop on Gesture in Human-Computer Interaction and Simulation, Lisbon, Portugal, May 2007.

112. Quinderé, M., Comunicação Humano-Robô através de Liguagem Falada. PhD Thesis, Universidade de Aveiro, May 2013

113. McKeown, K., Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text. Cambridge University Press, 1985

2 State of the Art on Elderly Speech

What is Elderly Speech?

Up until now, many disciplines have dealt with the subject of elderly people in various ways, e.g. Psychology, Social Psychology, Sociology and Gerontology. Therefore, the topic of elderly speech has established itself as a field on its own and is an interdisciplinary research area with diverse approaches of investigation, which can be proved by the disparity of studies and methodologies, which are, in most of the cases, extremely difficult to compare [1][2][3].

The existing articles draw a divergent picture about how to characterize elderly speech. While some propose criteria to distinguish between elderly versus teenagers' or adults' speech, there are others denying that there are clear-cut differences [4]. The absence of a single deterministic phonetic cue, existent, for example, in gender determination, makes elderly speech classification inexact. Since aging increases the difference between biological age and chronological age and considering that biological aging can be influenced by factors such as, abuse or overuse of the vocal folds, smoking, alcohol consumption, psychological stress/tension, or frequent loud/shouted speech production without vocal training [5][6] it is not possible to determine an exact age limit for speech to be considered as elderly. Conducted studies referenced in this document consider ages between 60 and 70 as the minimum age for the elderly age group [19].

Kohrt and Kucharczik [3] posed the question of if it is at all possible and crucial to determine the affiliation to certain, merely numerically determined age category for the linguistic competence and/or performance of their speech members. Cheshire followed the same line of thoughts [2] trying to come closer to a plausible answer, while investigating if it is reasonable to look for *'age markers'*. As a consequence she proposed the so called *"age-exclusive features"* and *"age-preferential features"* postulating that *"[t]he characteristic forms may be age-exclusive, in that they are used only during a certain stage of life, or they may be age-preferential, in that they occur more frequently in some stages of life than in others."*

Fiehler [7] follows the option that what is hastily called to be 'typical for the speech of elderly people' results from different situational circumstances, from which different registers are drawn, being this with respect to lexical and grammatical aspects. Though in contrast to teenagers' speech, which is often used as a social identifying characteristic, seniors do not look for acceptation of a group in the same way, because of their experience of life, their (acquired) social status, etc., and at the same time, because they are therefore not necessarily dependent of a linguistic assignment from a peer-group.

Observations considering the voice of elderly people have proved that it is possible to state differences between elderly speech and teenagers or adults speech on an acoustic phonetic level [1]. With increasing age there is a deprivation of chest voice, general changes in frequencies, in the voice quality and the timbres. Changes in the heights of vowel formant

frequencies particularly appear in older men, not only for biological reasons, but also because of social changes. Following Gerritson [9], Heinl-Hutschinson [10] and Helfrich [1] differences also occur while looking at the speech rate which is slower. Simultaneously more breaks, more speech errors and a lower volume of speech were detectable.

American studies [11][12][13] also conclude that elderly subjects in overall produce lesser morphemes per utterance as well as lesser utterances per minute. Additionally, they assume that the subjects, while aging, eliminate more often compulsory grammatical morphemes as well as articles and possessive pronouns. Furthermore, these studies agree that utterances get overall shorter with increased age, that seniors produce lesser correct verb tenses and also other correct morphological forms and that there is a tendency to monotonous grammatical constructions (e.g., because of avoiding the use of different grammatical forms) when compared to younger speakers. Kemper [15] could add up these findings while unravelling an age-dependent reduction of complex syntactic structures concerning written language.

Automatic Speech Recognition of Elderly Speech

As seen in the previous section, age is a key physiological characteristic of a speaker that must be considered to human-computer interfaces (HCIs) based on speech [14]. Although being a stable characteristic when compared with the awareness and emotional state of a speaker, age influences the performance of a recognition engine, as several parameters of the speech wave form are modified, such as fundamental frequency, jitter, shimmer and harmonic noise ratios [16].

Additionally, with age, the cognitive and perceptual abilities decrease [17][18]. Studies show that speech recognition accuracy for subjects with an age below 15 or above 70 decreases dramatically when using acoustic models specifically aimed at young to middle aged adults. Experiments demonstrate an error rate increase of 50% when comparing senior users with a middle age user group [19]. Other works confirm the same results as can be seen below.

On the other hand, in the work by Anderson et al [19], it was found that Speech Recognition engines trained with elderly specific acoustic data showed a significant decrease in the error rate when compared to engines trained with regular data. The collected corpus contained 79 hours of speech from 297 elderly speakers, with an average age of 79. The speakers age and minutes recorded distribution are shown in figure 1. Training an elderly only model yielded a WER of 42.1% against a WER of 54.6% when using a non-elderly model, with a significant higher WER for males. The test corpus consisted of 7.5 hours of speech produced by 40 elderly speakers.

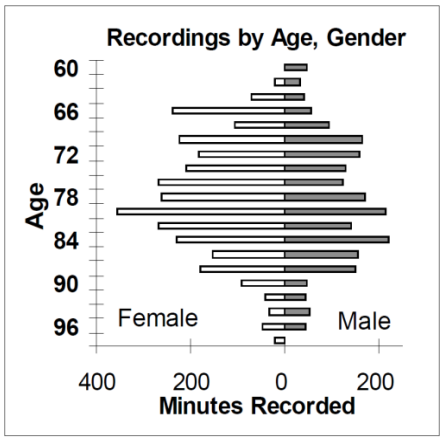

Figure 3 – Elderly speech recorded by gender and age

The model was used to assess the difference between regular queries to a document database using a keyboard and a speech. The results found that no significant difference was found in retrieval times or in the amount of help required between the two query modes, but the majority of the test subjects preferred the usage of speech, as it was perceived as a faster and easier interaction method.

Similar results were obtained in the work of Baba et al [21] where an improvement of 2.9% in the WER was achieved when training an acoustic model with an elderly only corpus. Similarly to the work of Anderson et al, a higher WER was found for male speakers. The used corpus consisted of 301 elderly speakers ranging from 60 to 90 years old, each one producing 200 utterances.

Very similar conclusion were obtained in the study by Vipperla et al. [22] where several experiments were conducted having as a basis the MATCH corpus [23]. This corpus consists of recordings from 24 young speakers (mean age was 22) and 26 elderly users (mean age 66) interacting with an automatic system for scheduling health care appointments. The total amount of recorded dialogs was 447. The authors detected several differences between the produced speech by the two subject groups. While the younger group restrained to simpler vocabulary and followed the instructions provided by the system, the older group often used richer vocabulary and tried to use the system in a more “Human to human” interaction method, not following the provided instructions.

In order to compensate for this difference in interaction style, two language models were created; one for each group, yielding some expected results: the performance of the language model built specifically for the younger group performed significantly worst when used by the older group.

Vipperla et al. [22] also performed experiments with acoustic modelling using the MATCH corpus and HTK, concluding that “older speech” performed significantly worse when used in a baseline model trained with other available speech corpora, as the MATCH corpus by itself

was insufficient to generate a new model from scratch. In actual numbers, the WER increases 11% when the test corpus is entirely formed by speech from the older group, in relation to a test corpus formed exclusively by speech from the young group.

In the work by Raux et al from Carnegie Mellon University, the "Let's go" [24] system was developed, where the main goal was to improve the quality of spoken dialogs in speech recognition systems for elderly and non-native, users that are not typically targeted for SR and TTS systems. The work focused in developing a system for informing users about the bus schedule network in Pittsburgh. The authors have found that while the non-native population shows difficulties in generating speech in the foreign language, the elderly population has difficulties in comprehending the information provided by the TTS system, where the degree of lack of comprehension increases with age. The used SR engine was CMU Sphinx [25] and for TTS, the Festival [26] system was used. However, no objective results were provided.

An interesting investigation work was conducted by Pieper and Kobsa [27], which focused in a particular study of a bed-ridden user that had no motor capabilities and had several speech difficulties due to the usage of an artificial ventilator system. The system consists essentially of a computer hooked into a video projector that displays an image on the ceiling right above the patient's bed. The system is able to recognize speech using Dragon's Dictate software [28], so the patient is effectively "talking to the ceiling". Several SR engine training procedures had to be taken in order to adapt to the peculiarities of the speaker's speech as well as to cancel the continuous hissing noise generated by the breathing apparatus.

Another paper that can be refereed is the work developed by Siohan et al. [29] having as basis the English subset of the MALACH (Multilingual Access to Large Audio archives) corpus. This corpus consists of 116,000 hours of digitized interviews in 32 languages from 52,000 survivors, liberators, rescuers and witnesses of the Nazi Holocaust, where the English subset used contains approximately 64 hours collected from 265 speakers within the age range of 55 to 95. Although the focus of the paper resided in noise compensation techniques for the low average SNR that the corpus shows, it was also concluded that speaker age is a relevant factor for the degradation of ASR systems when used within the elderly population group.

In summary, the analysed works show that, in overall, generic trained ASR systems perform significantly worse when used by the elderly population, due to various factors. The typical strategy to improve ASR performance under these cases is to collect speech data from elderly users in the specific domain of the target application and train elderly-only acoustic models.

Silent Speech Interfaces

An alternative way to perform ASR with elderly populations is to use the technique of Silent Speech Interface (SSI). Using this technique, a system can perform automatic speech recognition (ASR) in the absence of an intelligible acoustic signal and can be used as a human-computer interface (HCI) modality in high-background-noise environments such as living

rooms, or in aiding speech-handicapped individuals such as elderly persons. By acquiring sensor data from elements of the human speech production process – from the articulators of glottal activity, their neural pathways or the brain itself – an SSI produces a digital representation of speech which can be recognized and interpreted as data, synthesized directly or routed into a communications network.

The existent experimental SSI systems described in the literature are based on the following approaches: capture of the movement of fixed points on the articulators using Electromagnetic Articulography (EMA) sensors [30]; real-time characterization of the vocal tract using ultra-sound (US) and optical imaging of the tongue and lips [30][32][33][34][35]; digital transformation of signals from a Non-Audible Murmur (NAM) microphone (a type of stethoscopic microphone) [36][37][38][39]; analysis of glottal activity using electromagnetic [40][41], or vibration [42] sensors; surface electromyography (sEMG) of the articulator muscles or the larynx [43][44][45][46][47]; interpretation of signals from electro-encephalographic (EEG) sensors [48]; interpretation of signals from implants in the speech-motor cortex [49] or processing of signals from low power radar devices [50].

Silent speech interpretation through an electronic system or computer brought the attention of the community, as early as in 1968. The idea of lip-reading was spread by Stanley Kubrick's 1968 science-fiction film "2001 – A Space Odyssey", where a ''HAL 9000" computer was able to automatically lip-read the conversations [51]. It was only later that the first real solutions appeared. An example of this is the automatic visual lip-reading by Petajan [52], the patents registered for lip-reading equipment by Nakamura [53] and electromyography sensors developed by Sugie [54], in 1985, that achieved a 71% accuracy while recognizing 5 Japanese vowels. Working on a similar problem, Hasegawa [55] achieved, in 1991, a 91% recognition rate but this time using a video of the speaker's face where lip and tongue features were extracted. The idea of also recovering glottal excitation cues from voiced speech in noisy environments was focused by DARPA (Defense Advanced Research Projects Agency) with the Advanced Speech Encoding Program (ASE), at the early 2000's stimulating speech processing through the use of multiple mechanical and electromagnetic sensors [56][40][41].

With the massive adoption of cellular telephones around 1994 [51], SSIs started to appear as a possible solution for problems such as privacy in personal communications, and for users who had lost their capacity to produce voiced speech. In Japan, in 2002, with the possibility of robustness of silent speech devices in noisy environments, a NTT DoCoMo press release announced a prototype silent cellphone using EMG and optical capture of lip movement [57], specially targeting cellphone privacy. With the development of new sensing technologies and the advances made by the speech community, the ability to extract detailed real-time information about the human speech production process, has improved. Currently, technologies such as ultrasounds (US) [58,59,60]; X-ray cineradiography [61,62], fMRI [63,64], EMA [65,66], EMG [67,54], EPG [68] and Radar-like sensors [50], are being applied to silent speech interfaces related problems providing new possible approaches and ideas. There are

also several experiments on BCI techniques where SSI signals are explored at a brain level [69,70,71]. These latter types of SSIs mostly apply to people with disabilities, such as the locked-in syndrome [72].

The existent SSIs have been mainly developed by investigation groups from EUA [31], Germany [73], France [74] and Japan [36], and focused on their respective languages. There is no published work for European Portuguese in the area of SSIs, although there are previous investigations on related areas, such as: use of EMA [75], Electroglotograph and MRI [76] for speech production studies, articulatory synthesis [77] and multimodal interfaces involving speech [78,79].

Regarding elderly speech, observations considering the voice of elderly people have proved that it is absolutely plausible to state differences between elderly speech and teenagers' or adults' speech on an acoustic phonetic level [1]. With increasing age there is a deprivation of chest voice, general changes in frequencies, in the voice quality and the timbres. Changes in the heights of vowel formant frequencies particularly appear in older men, not only for biological reasons, but also because of social changes. Following Gerritson [80], Heinl-Hutschinson [81] and Helfrich [1] differences also occur while looking at the speech rate which is slower. Simultaneously more breaks, more speech errors and a humbled volume of speech were detectable.

References

[1] Helfrich, H. (1979): Age markers in speech. In: Scherer, K. & Giles, H.: Social markers in speech. Cambridge: University Press.

[2] Cheshire, J. (1987): Age and generation-specific use of language. In: Ammon, U., Dittmar, N. & Mattheier, K. (eds.): *Sociolinguistics. An international handbook of the science of language and society. First volume.* Berlin/New York: de Gruyter (= Handbücher zur Sprach- und Kommunikationswissenschaft).

[3] Kohrt, M. & Kucharczik, K. (2003): 'Sprache' – unter besonderer Berücksichtigung von 'Jugend' und 'Alter'. In: Fiehler, R. & Thimm, C.: *Sprache und Kommunikation im Alter*, pp. 17-37.

[4] Ryan, E. & Cole, R. (1990): Evaluative perceptions of interpersonal communication with elders. In: Giles, H., Coupland, N. & Wiemann, J. M. (eds.): Communication, health and the elderly. London: Manchester University Press, pp. 172-191.

[5] Linville, S.E.: Vocal Aging. Singular, San Diego (2001)

[6] Jessen, M.: Speaker Classification in Forensic Phonetics and Acoustics. In: Müller, C. (ed.) Speaker Classification I. LNCS(LNAI), vol. 4343, Springer, Heidelberg (2007)

[7] Fiehler, R. (1997): Kommunikation im Alter und ihre sprachwissenschaftliche Analyse. Gibt es einen Kommunikationsstil des Alters? In: Sandig, B. & Selting, M. (Hrsg.): Sprech- und Gesprächsstile. Berlin/New York: de Gruyter, pp. 345-370.

[8] Boden, D. & Bielby, D. D.V. (1983): The past as a resource. A conversational analysis of elderly talk. In: *Human Development*, 26, 308-319.

[9] Gerritson, M. (1985): Alters- und geschlechtsspezifische Sprachverwendung. In: Besch, W. & Mattheier, K. J. (eds.): *Ortssprachenforschung.* Beiträge zu einem Bonner Kolloquium. Berlin: Schmidt; 79-108.

[10] Heinl-Hutchinson, M. (1975): *Untersuchung zur Sprechweise und deren Beziehung zur Lebenszufriedenheit älterer Menschen.* Diplomarbeit, Universität Gießen.

[11] Kemper, S. & Kynette, D. (1986): Aging and the loss of grammatical forms: A cross-sectional study of language performance. In: *Language & Communication*, 6, 65-71.

[12] Light, L. L. (1993): Language changes in old age. In: Blanken, G. et al. (eds.): *Linguistic disorders and pathologies. An international handbook.* Berlin/New York: de Gruyter (= Handbücher zur Sprach- und Kommunikationswissenschaft, 8): 900-918.

[13] Stover, S. E. & Haynes, W. O. (1989): Topic manipulation and cohesive adequacy in conversations of normal adults between the ages of 30 and 90. In: *Clinical Linguistics & Phonetics*, 3: 137-149.

[14] Speaker Characteristics, Tanja Schultz, In: C. Müller (Ed.) Speaker Classification, Lecture Notes in Computer Science / Artificial Intelligence, Springer, Heidelberg - Berlin - New York, Volume 4343, 2007. To appear.

[15] Kemper, S. (1987): Life-span changes in syntactic complexity. In: *Journal of Gerontology*, 42: 323-328.

[16] Xue, S.A., Hao, G.J.: Changes in the human vocal tract due to aging and the acoustic correlates of speech production: a pilot study. Journal of Speech, Language, and Hearing Research 46, 689–701 (2003).

[17] Baeckman, L., Small, B. J., & Wahlin, A. (2001). Aging and memory: Cognitive and biological perspectives. In J. E. Birren & K. W. Schaie (Eds.), Handbook of the psychology of aging (pp. 349–377). San Diego, CA, USA: Academic Press.

[18] Fozard, J.L., & Gordon-Salant, S. (2001). Changes in vision and hearing with aging. In J. E. Birren & K. W. Schaie (Eds.), Handbook of the psychology of aging (pp. 241–266). San Diego, CA, USA: Academic Press.

[19] Wilpon, J. G., and Jacobsen, C. N., "A Study of Speech Recognition for Children and the Elderly", IEEE International Conference on Acoustics, Speech, and Signal Processing. Atlanta, May 1996, p. 349.

[20] Anderson, S., Liberman, N., Bernstein, E., Foster, S., Cate, E., Levin, B., and Hudson, R. 1999. Recognition of elderly speech and voice-driven document retrieval. In *Proceedings of the Acoustics,* Speech, and Signal Processing, 1999. on 1999 IEEE international Conference - Volume 01 (March 15 - 19, 1999). ICASSP. IEEE Computer Society, Washington, DC, 145-148. DOI=http://dx.doi.org/10.1109/ICASSP.1999.758083

[21]Baba Akira; Yoshizawa Shin'ichi; Yamada Miichi; Lee A; Shikano Kiyohiro; Elderly Acoustic Models for Large Vocabulary Continuous Speech Recognition; IEICE Transactions on Information and Systems, Pt.2 (Japanese Edition), 0915-1923, Vol.J85-D-2; No.3; Page 390-397, 2002.

[22]Vipperla, R., Wolters, M., Georgila, K., and Renals, S. 2009. Speech Input from Older Users in Smart Environments: Challenges and Perspectives. In Proceedings of the 5th international on Conferenceuniversal Access in Human-Computer interaction. Part Ii: intelligent and Ubiquitous interaction Environments (San Diego, CA, July 19 - 24, 2009). C. Stephanidis, Ed. Lecture Notes In Computer Science, vol. 5615. Springer-Verlag, Berlin, Heidelberg, 117-126. DOI=http://dx.doi.org/10.1007/978-3-642-02710-9_14

[23]Kallirroi Georgila, Maria Wolters, Johanna D. Moore, Robert H. Logie; The MATCH corpus: a corpus of older and younger users' interactions with spoken dialogue systems; Language Resources and Evaluation; DOI: 10.1007/s10579-010-9118-8

[24]A. Raux, B. Langner, A. Black, and M. Eskenazi, "LET'S GO: Improving spoken dialog systems for the elderly and non-native," in Eurospeech03, Geneva, Switzerland, 2003.

[25]X. Huang, F. Alleva, H.-W. Hon, K.-F. Hwang, M.-Y. Lee, and R. Rosenfeld, "The SPHINX-II speech recognition system: an overview," Computer Speech and Language, vol. 7(2), pp. 137–148, 1992.

[26]A. Black, P. Taylor, and R. Caley, "The Festival speech synthesis system," http://festvox.org/festival, 1998.

[27]Michael Pieper , Alfred Kobsa, Talking to the ceiling: an interface for bed-ridden manually impaired users, CHI '99 extended abstracts on Human factors in computing systems, May 15-20, 1999, Pittsburgh, Pennsylvania DOI: 10.1145/632716.632723.

[28]Nuance, Dragon NaturallySpeaking Solutions, http://www.nuance.com/naturallyspeaking/, last visited on 26-03-2010.

[29]Siohan, Olivier; Ramabhadran, Bhuvana; Zweig, Geoffrey (2004): "Speech recognition error analysis on the English MALACH corpus", In INTERSPEECH-2004, 413-416.

[30]Fagan, M.J., Ell, S.R., Gilbert, J.M., Sarrazin, E., Chapman, P.M., 2008. Development of a (silent) speech recognition system for patients following laryngectomy. Med. Eng. Phys. 30 (4), 419–425.

[31]Thomas Hueber, Elie-Laurent Benaroya , Gérard Chollet, Bruce Denby, Gérard Dreyfus, Maureen Stone, Visuo-Phonetic Decoding using Multi-Stream and Context-Dependent Models for an Ultrasound-based Silent Speech Interface, in Proceedings of Interspeech 2009, Brighton, UK; September 2009.

[32]Denby, B., Stone, M., 2004. Speech synthesis from real time ultrasound images of the tongue. In: Proc. IEEE Internat. Conf. on Acoustics, Speech, and Signal Processing, (ICASSP'04), Montre´al, Canada, 17–21 May 2004, Vol. 1, pp. I685–I688.

[33]Hueber, T., Aversano, G., Chollet, G., Denby, B., Dreyfus, G., Oussar, Y., Roussel, P., Stone, M., 2007a. Eigentongue feature extraction for an ultrasound-based silent speech interface. In: IEEE Internat. Conf. on Acoustic, Speech, and Signal Processing, ICASSP07, Honolulu, Vol. 1, pp. 1245–1248.

[34]Hueber, T., Chollet, G., Denby, B., Dreyfus, G., Stone, M., 2008a. Phone Recognition from Ultrasound and Optical Video Sequences for a Silent Speech Interface. Interspeech, Brisbane, Australia, pp. 2032-2035. Hueber, T., Chollet, G., Denby, B., Stone, M., 2008b. Acquisition of ultrasound, video and acoustic speech data for a silent-speech interface application. In: Internat. Seminar on Speech Production, Strasbourg, France, pp. 365–369.

[35]Hueber, T., Benaroya, E.L., Chollet, G., Denby, B., Dreyfus, G., Stone, M., (2009) "Development of a Silent Speech Interface Driven by Ultrasound and Optical Images of the Tongue and Lips", Speech Communication, à paraître (doi:10.1016/j.specom.2009.11.004).

[36]Tomoki Toda, Keigo Nakamura, Takayuki Nagai,Tomomi Kaino, Yoshitaka Nakajima, Kiyohiro Shikano, Technologies for Processing Body-Conducted Speech Detected with Non-Audible Murmur Microphone, in Proceedings of Interspeech 2009, Brighton, UK; September 2009.

[37]Nakajima, Y., 2005. Development and evaluation of soft silicone NAM microphone. Technical Report IEICE, SP2005-7, pp. 7–12 (in Japanese).

[38]Nakajima, Y., Kashioka, H., Campbell, N., Shikano, K., 2006. Non-audible murmur (NAM) recognition. IEICE Trans. Inform. Systems E89-D (1), 1–8.

[39]Tran, V.-A., Bailly, G., Loevenbruck, H., Toda, T., 2008b. Predicting F0 and voicing from NAM-captured whispered speech. In: Proc. Speech Prosody, Campinas, Brazil.

[40]Tardelli, J.D. (Ed.), 2003. MIT Lincoln Labs Report ESC-TR-2004-084. Pilot Corpus for Multisensor Speech Processing.

[41]Quatieri, T.F., Messing, D., Brady, K., Campbell, W.B., Campbell, J.P., Brandstein, M., Weinstein, C.J., Tardelli, J.D., Gatewood, P.D., 2006. Exploiting non-acoustic sensors for speech enhancement. IEEE Trans. Audio Speech Lang. Process. 14 (2), 533–544.

[42]Patil, S.A., Hansen, J.H.L., this issue. A competitive alternative for speaker assessment: physiological Microphone (PMIC). Speech Comm.

[43]Yunbin Deng, Rupal Patel, James T. Heaton, Glen Colby, L. Donald Gilmore, Joao Cabrera, Serge H. Roy, Carlo J. De Luca, Geoffrey S. Meltzner, Disordered Speech Recognition Using Acoustic and sEMG Signals, in Proceedings of Interspeech 2009, Brighton, UK; September 2009.

[44]Michael Wand, Szu-Chen Stan Jou, Arthur R. Toth, Tanja Schultz, Synthesizing Speech from Electromyography using Voice Transformation Techniques, in Proceedings of Interspeech 2009, Brighton, UK; September 2009.

[45]Maier-Hein, L., Metze, F., Schultz, T., Waibel, A., 2005. Session independent non-audible speech recognition using surface electromyography. In: IEEE Workshop on Automatic Speech Recognition and Understanding, San Juan, Puerto Rico, pp. 331–336.

[46]Jou, S., Schultz, T., Walliczek, M., Kraft, F., 2006. Towards continuous speech recognition using surface electromyography. In: INTERSPEECH 2006 and 9th Internat. Conf. on Spoken Language Processing, Vol. 2, pp. 573–576.

[47]Charles Jorgensen, Sorin Dusan, Speech interfaces based upon surface electromyography, Speech Communication, Volume 52, Issue 4, Silent Speech Interfaces, April 2010, Pages 354-366, ISSN 0167-6393, DOI: 10.1016/j.specom.2009.11.003.

[48]Porbadnigk, A., Wester, M., Calliess, J., Schultz, T., 2009. EEG-based speech recognition – impact of temporal effects. In: Biosignals 2009, Porto, Portugal, January 2009, pp. 376–381.

[49]Brumberg, J.S, Nieto-Castanon, A., Kennedy, P.R., Guenther, F.H., this issue. Brain–computer interfaces for speech communication. Speech Comm.

[50]John F. Holzrichter, Characterizing Silent and Pseudo-Silent Speech using Radar-like Sensors, in Proceedings of Interspeech 2009, Brighton, UK; September 2009.

[51]Denby, B. et al., Silent speech interfaces, Speech Comm. (2009), doi:10.1016/j.specom.2009.08.002

[52]Petajan, E.D., 1984. Automatic lipreading to enhance speech recognition. In: IEEE Communications Society Global Telecommunications Conf., Atlanta, USA.

[53]Nakamura, H., 1988. Method of recognizing speech using a lip image. United States Patent 4769845, September 06.

[54]Sugie, N., Tsunoda, K., 1985. A speech prosthesis employing a speech synthesizer-vowel discrimination from perioral muscle activities and vowel production. IEEE Trans. Biomed. Eng. BME-32 (7), 485–490.

[55]Hasegawa, T., Ohtani, K., 1992. Oral image to voice converter, image input microphone. In: Proc. IEEE ICCS/ISITA 1992 Singapore, Vol. 20, No. 1, pp. 617–620.

[56]Ng, L., Burnett, G., Holzrichter, J., Gable, T., 2000. Denoising of human speech using combined acoustic and EM sensor signal processing. In: Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), Istanbul, Turkey, 5–9 June 2000, Vol. 1, pp. 229–232.

[57]Fitzpatrick, M., 2002. Lip-reading cellphone silences loudmouths, New Scientist, edition of 03 April 2002.

[58]Stone, M., Davis, E., 1995. A head and transducer support (HATS) system for use in ultrasound imaging of the tongue during speech. J. Acoust. Soc. Amer. 98, 3107–3112.

[59]Stone, M., 2005. A guide to analyzing tongue motion from ultrasound images. Clin. Linguist. Phonet. 19 (6–7), 455–502.

[60]Wrench, A., Scobbie, J., Linden, M., 2007. Evaluation of a helmet to hold an ultrasound probe. In: Ultrafest IV, New York, USA.

[61]Arnal, A., Badin, P., Brock, G., Connan, P.-Y., Florig, E., Perez, N., Perrier, P. Simon, P., Sock, R., Varin, L., Vaxelaire, B., Zerling, J.-P., 2000. Une base de donne´es cine´radiographiques du fran_cais, XXIIIe`mes Journe´es d'Etude sur la Parole, Aussois, France, pp. 425–428.

[62]Munhall, K.G., Vatikiotis-Bateson, E., Tohkura, Y., 1995. X-ray film database for speech research. J. Acoust. Soc. Amer. 98, 1222–1224.

[63]Gracco, V.L., Tremblay, P., Pike, B., 2005. Imaging speech production using fMRI. NeuroImage 26 (1), 294–301, 15 May.

[64]NessAiver, M.S., Stone, M., Parthasarathy, V., Kahana, Y., Paritsky, A., 2006. Recording high quality speech during tagged cine-MRI studies using a fiber optic microphone. J. Magnet. Reson. Imag. 23, 92–97.

[65]Perkell, J., Cohen, M., Svirsky, M., Matthies, M., Garabieta, I., Jackson, M., 1992. Electro-magnetic midsagittal articulometer (EMMA) systems for transducing speech articulatory movements. J. Acoust. Soc. Amer. 92, 3078–3096.

[66]Hoole, P., Nguyen, N., 1999. Electromagnetic articulography in coarticulation research. In: Hardcastle, W.H., Hewlitt, N. (Eds.), Coarticulation: Theory, Data and Techniques. Cambridge University Press, pp. 260–269.

[67]Tatham, M., 1971. The place of electromyography in speech research. Behav. Technol. 6.

[68]Bibliography of Electropalatographic (EPG) Studies in English (1957–2005), Queen Margaret University, Edinburgh, UK, September 2005. <http://www.qmu.ac.uk/ssrc/cleftnet/EPG_biblio_2005_september.pdf>.

[69]Brain Computer Interfaces, IEEE Computer, Vol. 41, No. 10, October 2008.

[70]Sajda, P., Mueller, K.-R., Shenoy, K.V. (Eds.), 2008. Brain Computer Interfaces. IEEE Signal Process. Mag. (special issue).

[71]Epstein, C.M., 1983. Introduction to EEG and evoked potentials. J.B. Lippincot Co..

[72] Jonathan S. Brumberg, Philip R. Kennedy, Frank H. Guenther, Artificial speech synthesizer control by brain-computer interface, in Proceedings of Interspeech 2009, Brighton, UK; September 2009.

[73]Calliess, J.-P., Schultz, T., 2006. Further Investigations on Unspoken Speech. Studienarbeit, Universita¨ t Karlsruhe (TH), Karlsruhe, Germany.

[74]Viet-Anh Tran, Gérard Bailly, Hélène Loevenbruck , Tomoki Toda, Multimodal HMM-based NAM-to-speech conversion, in Proceedings of Interspeech 2009, Brighton, UK; September 2009.

[75]ROSSATO, Solange; TEIXEIRA, António; FERREIRA, Liliana - Les Nasales du Portugais et du Français : une étude comparative sur les données EMMA . In XXVI Journées d'Études de la Parole. Dinard, FR, Jun. 2006.

[76]MARTINS, Paula; CARBONE, Inês; PINTO, Alda; SILVA, Augusto; TEIXEIRA, António - European Portuguese MRI based speech production studies. Speech Communication. NL: Elsevier. ISSN: 0167-6393, vol. 50, nº 11/12 (12. 2008). p. 925 – 952.

[77]António Teixeira and Francisco Vaz, Síntese Articulatória dos Sons Nasais do Português, Anais do V Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada (PROPOR) 2000, pp. 183-193, ICMC-USP, Atibaia, São Paulo, Brasil.

[78]TEIXEIRA, António J. S.; MARTINEZ, Roberto; SILVA, Luís Nuno; JESUS, Luís M. T.; PRÍNCIPE, José Carlos; VAZ, Francisco A. C. - Simulation of Human Speech Production Applied to the Study and Synthesis of European Portuguese. Eurasip Journal on Applied Signal Processing: Hindawi Publishing Corporation, vol. 2005, nº 9 (Jun. 2005). p. 1435-1448.

[79]M. Sales Dias et al., Using Hand Gesture and Speech in a Multimodal Augmented Reality Environment, GW2007, LNAI 5085, pp.175-180, 2009.

[80]Gerritson, M. (1985): Alters- und geschlechtsspezifische Sprachverwendung. In: Besch, W. & Mattheier, K. J. (eds.): Ortssprachenforschung. Beiträge zu einem Bonner Kolloquium. Berlin: Schmidt; 79-108.

[81]Heinl-Hutchinson, M. (1975): Untersuchung zur Sprechweise und deren Beziehung zur Lebenszufriedenheit älterer Menschen. Diplomarbeit, Universität Gießen

3 Assistive Technologies for Seniors

The existent companies and organizations that develop applications for seniors usually use elderly targeted places such as, nursing homes, government subsidized housing, retirement communities, senior's centres and public libraries to develop, test and disseminate their software. The main objective of these applications is to provide access to web contents, to enable communication with friends and family, promote literacy, and overcome elderly reluctance towards media and electronic devices. According to [1], the number of seniors connected to the internet is rising hastily being the fastest-growing demographic group online. Internet is also increasingly becoming an important resource for information about health and health care options, communication and news. By being connected to the outside world, senior citizens become more socially integrated and have fewer depressive symptoms [2]. The applications for seniors are characterized by friendly interfaces with buttons and font sizes above normal in order to tackle with low vision issues that are characteristic of the population in this age group. There are also applications for tracking and surveillance that can be applied to seniors that require monitoring [8]0.

As stated in [11] speech can also be applied has an HCI in software that targets the elderly age group. The examples shown in this document make use of Spoken Dialogue systems to increase the level of interaction with technology.

In the following sections, we describe several examples of applications that target seniors.

Speech-enabled Accessibility Applications

This section presents accessibility applications currently available in the market, targeting seniors and other target users that have a speech interface.

Windows accessibility features

The OS Windows from Microsoft has several accessibility features built-in specially designed for users with special needs. These features focus essentially in improvising screen readability, as well as improving and facilitating the way users interact with the PC. The key accessibility features from Windows follow:

- **Speech Commanding and Dictation:** For users who do not utilize the keyboard or mouse or who prefer using speech for dictation, Windows includes a Speech function that enables the user to dictate documents (including e-mail), navigate the Internet, and command applications and the operating system. Windows also adds the capability to dictate into almost any application.
- **High Dots Per Inch ("DPI"):** This feature enables user to scale the user interface to make text and graphics easier to see while preserving the quality of the users' viewing experience. The High DPI setting is a per-user setting, which allows for a PC with multiple users to have a personalized setting.

- **Magnifier:** Windows Magnifier enlarges portions of the screen. This is especially useful for viewing objects that are difficult to see, but also for seeing the whole screen more easily. Magnifier includes the new capability to magnify the entire desktop, and includes a new lens mode that allows a user to magnify a portion of the screen. A screenshot of the magnifier tools follows:

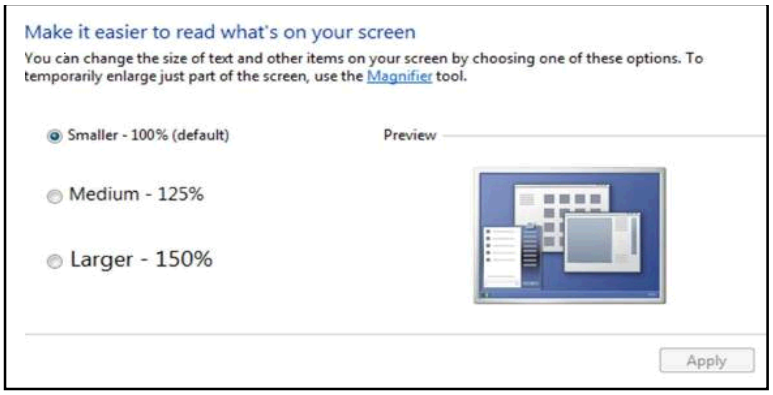

Figure 4 - Windows magnifier tool

- **Narrator:** Narrator is a text-to-speech program (or basic screen reader) that is built into Windows. Narrator reads on-screen menus to help users control the computer. This basic screen reader may work for the casual computer user.
- **On-Screen Keyboard**: On-Screen Keyboard ("OSK") can be resized to make it easier to see and use (including by highlighting or "glowing" certain keys), and includes text prediction in eight languages, which speeds up typing. The OSK also includes a scanning feature that allows people who use alternate input devices to click the on-screen keys. A screenshot follows

Figure 5 - On-Screen Keyboard

QualiWorld platform

QualiWorld is a software platform that enables disabled and elderly persons to access and use a computer [18]. The platform manages several applications (QualiWord, QualiMail, QualiRadio, QualiSurf, etc.) that interact with each other. The software allows the user to accomplish simple daily tasks such as, writing a letter, preparing a document, communicating verbally, surfing the Internet, sending and reading e-mail messages, making phone calls and controlling your household environment and watching a movie. The program is available in English, Italian, French and German. QualiWorld also interacts with TTS engines and ASR systems as alternative interfaces.

QualiWorld provides several accessibility solutions that can replace a physical mouse and keyboard. The appearance of the software can also be personalized. The application can be tailored, at any time, to specific user's ability and needs. Below are listed the accessibility solutions provided.

Mouse control solutions:

- Auto-Scan: items on the computer screen are sequentially and automatically highlighted, one after the other. When the desired function is highlighted, the user activates the switch (or left mouse button) to make his selection. User can manage the click of the mouse by any external switch.
- Manual-Scan: items on the computer screen are sequentially highlighted by pressing on the switch. The user can make his selection by keeping the switch pressed (scan with 1 switch) or by activating a second switch (scan with 2 switches).
- Radar Mouse: a coloured line (from the center to the border of the screen) scans the computer screen (360°). With any external switch, the user can stop the line with a click. An animated cursor starts moving from the center of the screen following the line. When the cursor reaches the desired function (intersection), user activates the switch and makes his selection.
- XY Mouse: a horizontal line scans the computer screen from top to bottom. When the line reaches the height of desired button, the user activates the switch and the line stops moving. A vertical line starts the scanning from left to right, and when it reaches the desired intersection point (button), the user can activate the switch to make his selection.
- Direction Mouse: the cursor is moved using one of the 8 arrows pointing in different directions (up, down, left, right, 4 oblique directions). Arrows are highlighted sequentially, user clicks on the desired direction and cursor begins moving. A second click stops the cursor.
- Tracking Mouse: the cursor on the screen is controlled by simple head movements. A standard USB WebCam captures user's movements and the software translates them into onscreen cursor movement, in real time. Users do not have to attach anything to their body.

Mouse Click control solutions:

- Auto-click: stops the cursor on the desired function or command button and the click is automatically activated by the system.
- Gesture Recognition: use a body gesture to perform the click on the selected button (Tracking Mouse).

Onscreen Keyboard:

- QualiKEY application: is an onscreen keyboard that includes a layout editor, a multi-language word prediction system with vocabulary editor, abbreviations, macros, etc.

Figure 6 – QualiLife platform interface

NIHSeniorHealth Website

The website from the National Institutes of Health (NIH) contains basic health and wellness information for older adults. In order to increase the website accessibility for senior's controls to increase text size, change contrast or allow the user to hear the text read aloud were included. In the "read text aloud" feature the user just needs hoover the mouse over a link or an element that contains a tooltip. When the feature is turned on, a description paragraph at the top of the page is read. The figure below shows the top part of the page.

Figure 7 – NIHSeniorHealth webpage

Verbose Text to Speech

This goal of this application is to assist in listening to text by reading aloud any text and then saving it as mp3 or wav file for future listening. This application can be used by the elderly with low vision, slow reading or reading disability issues. The software uses Microsoft Sam for Text to Speech synthesis by default but also supports third party voices SAPI complaint. [7]. A screenshot of this application in action follows:

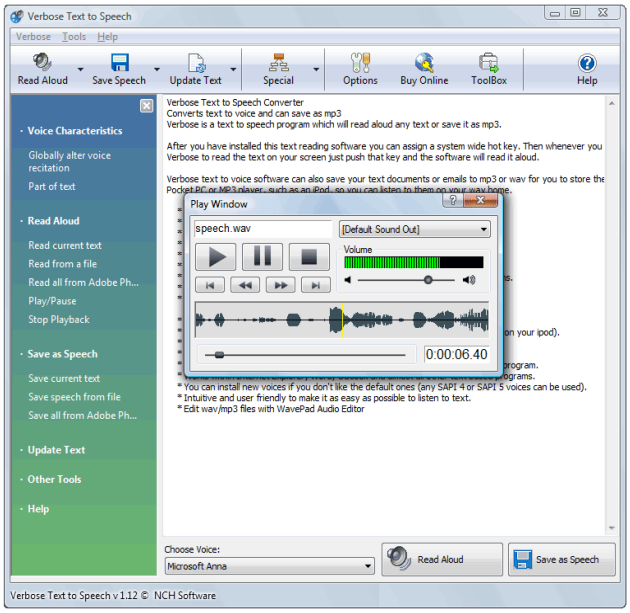

Figure 8 - Verbose text to speech application

I2net - Orion

Australia's i2net Computer Solutions has developed voice-recognition systems for the elderly. The company's Orion system uses Dragon Naturally Speaking voice recognition software [10] to activate household "smart-wired" appliances. The spoken word can be used to turn on a TV or the lights in a room. The system has also the ability to monitor network, so that a member of the subjects family can log on to check the actions performed such as, checking if the lights were turned on. To automate a house with the systems required technology costs at least $5000, according to the managing director of I2net [11].

Claro software – Lightning with Speech

Lightning with Speech software provides two categories of features. The first is related with visual aid and can be used by seniors to magnify the screen, change shapes and sizes of pictures, improve contrast. The second set of features provides a speech synthesizer for all kinds of text applications, such as WordPad, Skype, anti-virus programs, etc., and to navigate the web.

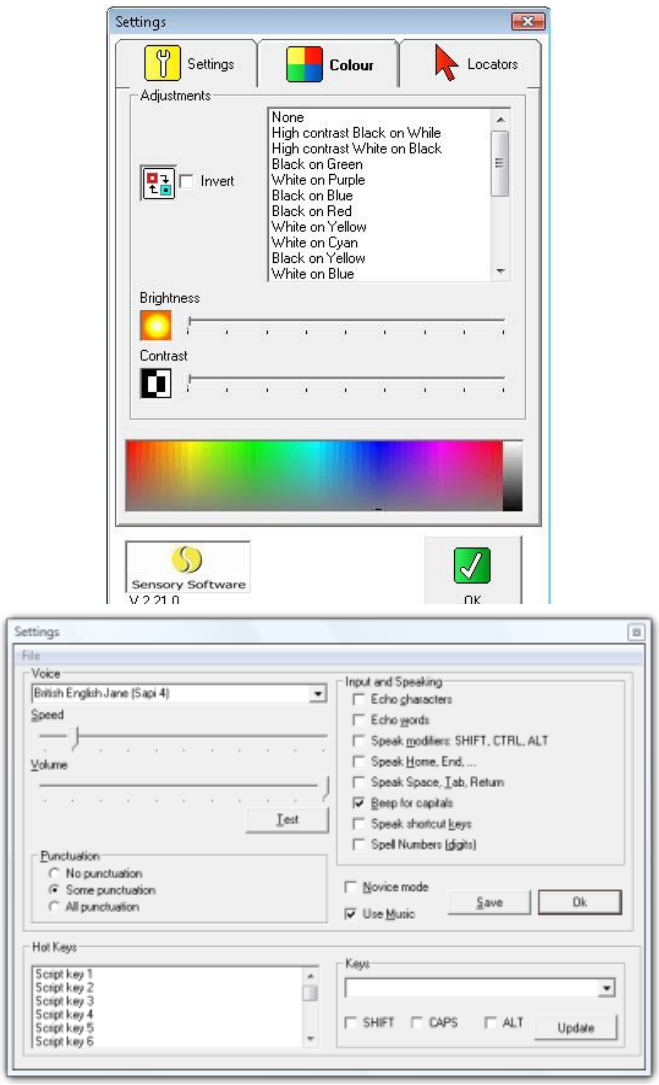

Figure 9 - Lightning with Speech interface

Non-Speech Accessibility Applications

This section presents some of the available market applications for seniors without a speech interface.

Doro

Doro is an international player, one of the most important leaders on the senior's market in the new technologies. It is present in Austria, Belgium, Denmark, Finland, France, Germany, Ireland, Italy, Luxemburg, Netherlands, Norway, Spain, Sweden, Switzerland, United Kingdom, Canada, United States, Australia, and New Zealand.

Doro is specialized in digital phones for the elderly. A recent press release shows that Doro dominates this segment of the international by selling 4 million cell phones worldwide. However, in the smartphone and apps market Doro is a new comer. As a matter of fact, the smartphone and apps market for the elderly is a young branch- no company dominates this market.

Doro's application for tablets and PCs consists in a simplified interface that provides easy access to internet and basic features such as the calendar or the photo gallery.

These apps are basic-communication oriented, unlike PLA which goes beyond the basic-communication orientation by adopting a social orientation.

Figure 8 - Doro's interfaces for the PC and the tablet apps

Generations on Line

Generations on Line is a software program that provides step-by-step instruction to help seniors use the internet. The objective of this program is to enhance communication amongst generations by promoting Internet access and literacy to seniors [3].

The following screenshots depict some of the functionalities provided by the "generations on line" platform

Figure 10 - Generations online start screen and "generation to generation" screen

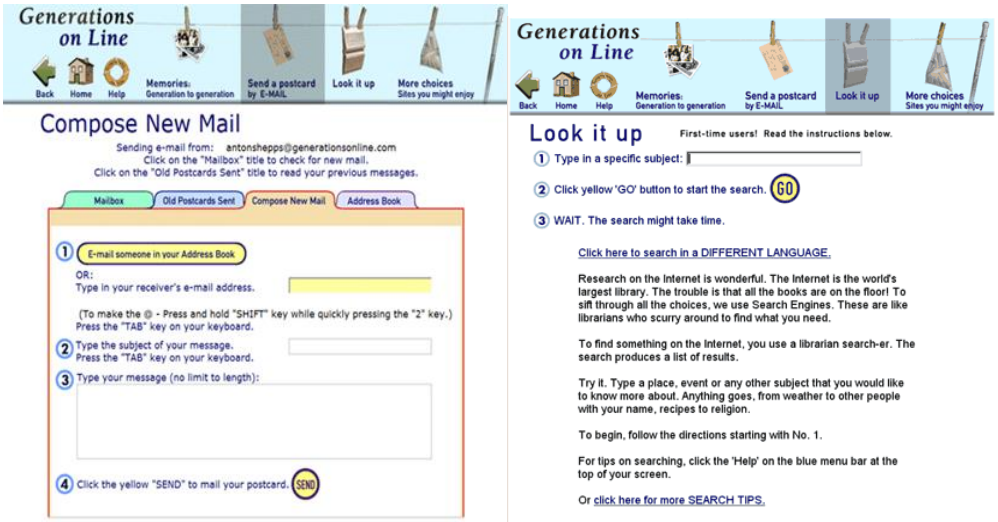

Figure 11 - Generations online being used to send email or search the net

PointerWare

PointerWare is user-friendly software especially designed for seniors. This platform allows performing basic computer tasks such as, managing email, play games, view photos or access the internet [4].

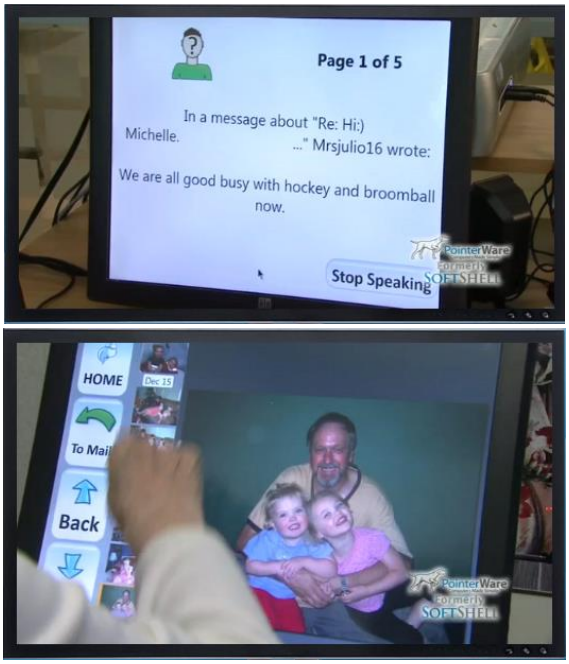

Figure 12 - Chatting or watching family pictures using PointerWare

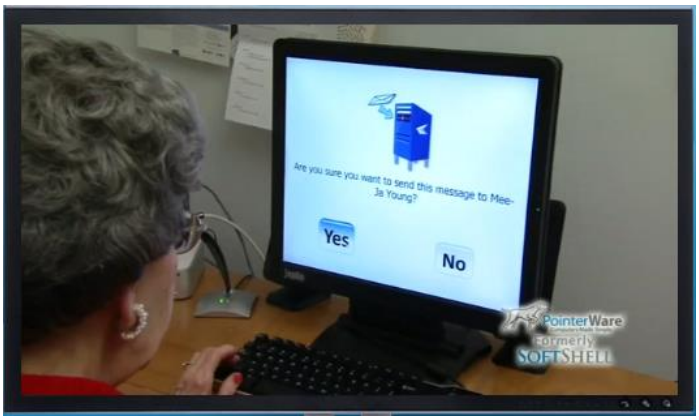

Figure 13 - Sending emails with PointerWare

Eldy

Eldy is a similar solution to PointerWare that provides seniors with an easier access to email, online chats (skype), weather forecasts, view photos or even watch television on the

computer. As a non-profit organization Eldy is free software and supported by volunteers [5]. Some screenshots follow.

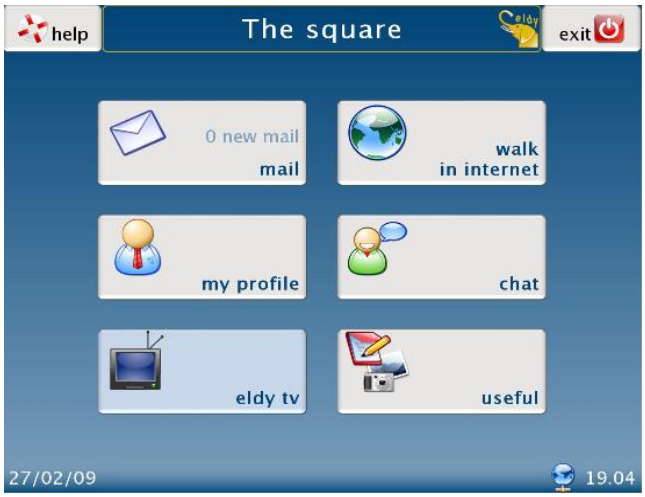

Figure 14 - Eldy start screen

Figure 15 - Eldy "useful" tools menu

Figure 16 - Watching photos using Eldy

IBS Diary

This software allows the user to record daily meals and medications. It also allows trying to find allergies, determine eating habits and bowel movements. The application is not only suitable for seniors but also for people with irritable bowel syndrome, bowel diseases or people with a low immune system. All the information is deliverable to the user's doctor, specialist or nurse [[6]. In the following picture, a screenshot of the IBS diary application can be seen.

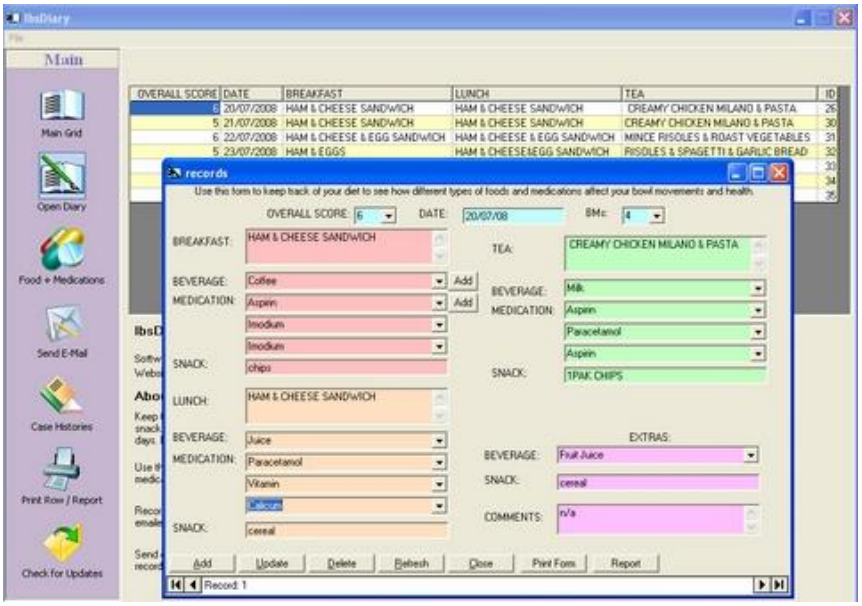

Figure 17 IBS diary application

Babysitter and Senior Caregiver

This tool allows the planning of events related with caregivers. It can be used by nannies, nurses, child specialists, elder caregivers and elder companions, as it provides them with an application to organize information about the subjects/clients. The tool can keep track of scheduled sitting appointments or personal information such as food allergy notes, medical info, comments and pictures. The application also supports to export the stored information into several formats such as, XLS, TXT or XML and database backups [8]. A screenshot follows:

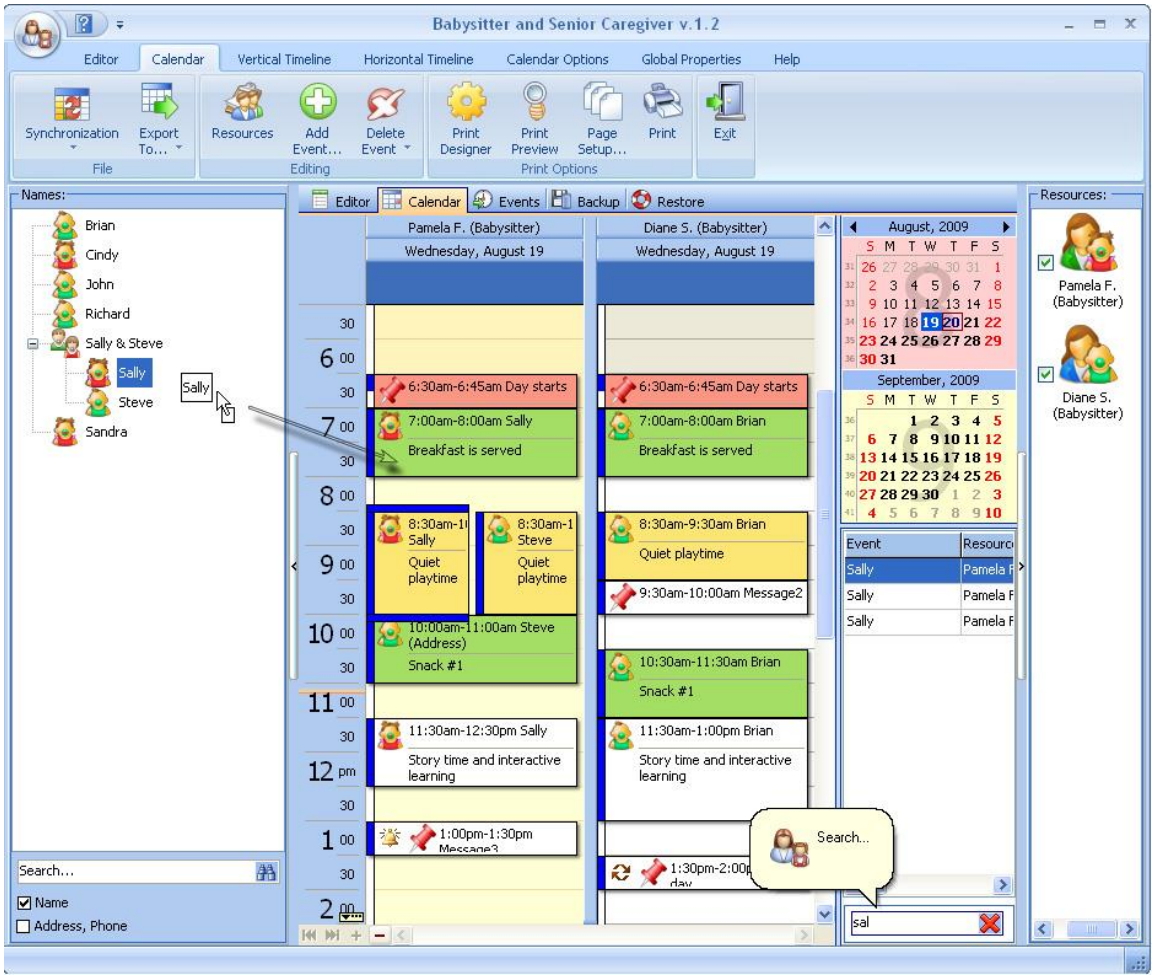

Figure 18 - The caregiver application

SatTracx

SacTracx is a service that allows knowing the location of a determined person or asset. The application is installed on a mobile device and through GPS determines the location of the marked entity. This service can apply to monitor seniors that require constant supervision 0.

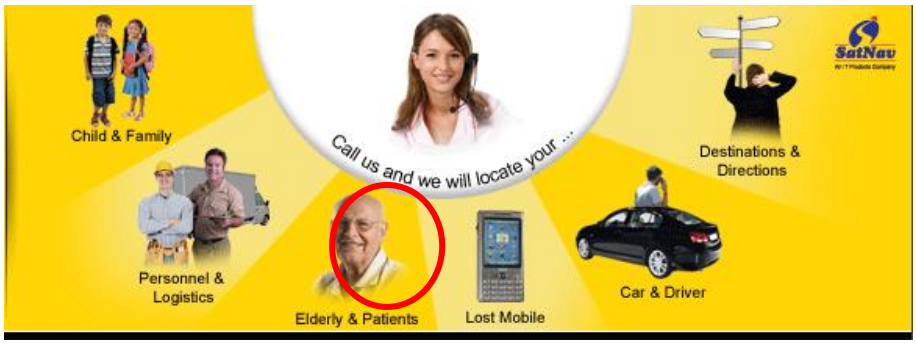

Figure 19 - SatTracx start screen

OnTimeRx

OnTimeRx is a medication reminder available as software (Blackberry, Pocket PC, PalmOS and Windows Desktop) and as a service (SMS, Phone and Email). The application allows a user to set up a reminder schedule and personalize messages. Thus, depending on the platform the user will be notified on his/her phone, email, desktop, etc [12].

Figure 20 – OnTimeRx Home form (left) and MyMeds form (right)

Figure 21 – OnTimeRx medical reminder and supply

Accessibility for seniors: full packages for a full service

We notice that the actual trend on the new technologies for seniors market is to offer packages which contain besides the software or the application associated services and devices. Many of these packages include a monthly fee instead of a one-time payment and some offer both of these possibilities.

Software as a service & full package offer

In this particular type of package we can observe that the accessibility is the combination of at least two elements. Usually, the accessibility of the interface combines with **the accessibility of the hardware characteristics and/or design** or with **the accessibility to internet, mobile services and assistance services**.

Ordissimo

The company that suits the first case is Ordissimo which provides hardware with preinstalled software and **accessories** such an external disk, a keyboard which contains direct commands (i.e.: a "Print" key) or a printer which can be installed and customized directly by the company.

Figure 22 – Ordissimo market offer

Ordissimo focuses on the hardware and the basic functions of a computer: writing texts or using the internet via basic functions (email, research…). Ordissimo's also focuses on the **assistances services**.

Although it offers the possibility of communicating via instant messaging, Ordissimo doesn't seem a communication-oriented product, nor a social one.
The interaction is a non-speech one. The interaction with Ordissimo's products can be done either by touch, keyboard and/or mouse (the case of all-in-one computer), by keyboard and mouse (Oridissimo's laptops) or by touch (Ordissimo's tablet).

Tooti Family: partial speech interaction (dictation module) & hardware

The Tooti Family's (France based company) offer can be characterized as a full package offer. In fact, the company offers different combinations of hardware, software, internet connection and assistance services that are available on a fix price and an additional monthly fee. For example, at 379€ +29,99€/month, the senior gets a tablet with the pre-installed software, mobile internet, assistance services and some features and premium services provided by Tooti Family.

Figure 23 - Tooti Family market offer

Here's the general presentation of Tooti Family' services and features:

- ***Tooti Features & Premium Services ("basic services" don't exist)***
    - **Tablet Features**: instant messaging, contacts, photos and videos, agenda, info, internet, games, tools, video call (wifi connection required);
    - **Family Features**: access to "family site", remote desktop, video call (wifi connection required);
    - **Premium Services**: Software &contents updates, anti-virus, anti-spam.

- ***Assistance Services (accessible by the senior or other persons delegated by the senior)***
    - "First steps" support;
    - Resolve possible problems;
    - Automatic saving of the data & restore of the data if needed.

- ***Mobile Internet Connection Services***
    - SIM card included, with 500 Mo, 50 sms/ month (to send), unlimited sms (to receive).

We note that the assistance is made via a hotline. Tooti Family doesn't offer home interventions or private lessons. Still, at Nantes thanks to a partnership with a social centre seniors can be assisted by the personnel of the social centre.
Besides this interesting partnership with the social centre in Nantes, personal online store, senior specialized retailers (*l'Univers du Confort- "The Universe of Comfort")*, informatics services providers & specialized providers for seniors informatics assistance such as *Facile&co* or *Docteur Ordinateur ("Computer Doctor")* are also Tooti Family's partners.

The images below offer a quick view on Tooti Family's interface:

Figure 24 - Tooti Family Interface

Non-speech interaction: products that use Kinect interaction

One of the actual trends in the field of human-machine interaction is the Kinect interaction. As a matter of fact the Kinect is integrated by many products as a tool in physical therapies designed as serious-games. A serious-game is a type of game which its purpose goes beyond entertainment. In the case of the serious-games that use the Kinect interaction their objective is the rehabilitation of persons that experience troubles in the coordination of their movements.

Some of the retirement homes are already using the Kinect and the Xbox games (such as bowling) in their physical exercise sessions as the pictures here below show:

Figure 25 - Elderlys playing Xbox games using Kinect

Also the Microsoft Kinect is used in a retirement home in the US as a monitoring tool that can detect eventual falls of the seniors or other behaviors that might be signs of medical problems.

In the pictures here below you can find the visual presentations of two products (The Voracy Fish serious game and Fovea Interactive) that combine the medical surveillance and recommendations with the individual practice at home.

Figure 26 - Voracy Fish

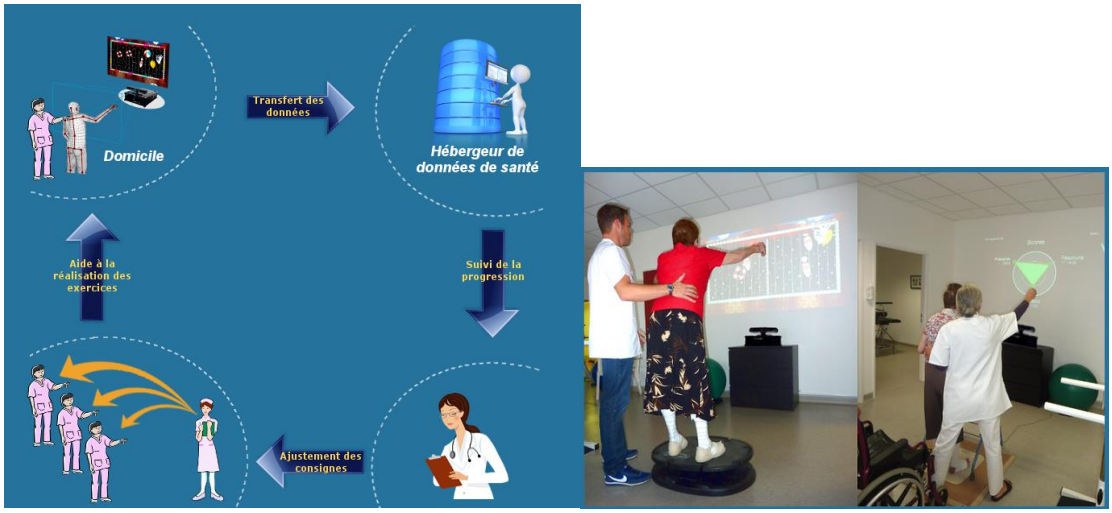

Figure 27 - Fovea Interactive: Evaluation & Recommendation Tool

The increasing popularity of Kinect interaction among the elderly is an advantage for PLA. Although PLA will not be sold as a tool for ergotherapy, Kinect interaction will not run the risk to be seen as a non-useful interaction or difficult to be used. Practically, seniors will know how it works and therefore it will be easier to convince them that interacting with PLA application via the Kinect will make interaction more accessible.

Accessibility Hardware

This section presents several categories of hardware that can be used by seniors.

Activo PC Sénior

Activo PC Sénior is an initiative of Microsoft, Caixa Geral de Depósitos, Rutis and Inforlândia in the scope of the digital literacy program created by Microsoft in Portugal. This initiative aims to bring the benefits of IT and the Internet communications to the community of Portuguese senior citizens, which already represents around 16% of the population (Census 2001 data).

The Activo PC Sénior is essentially a special designed laptop PC with the following features: a lighting device for the keyboard, enhanced keyboard with larger keys and larger spacing between keys, a wireless BT ergonomic designed mouse and a built-in 3G modem for mobile internet.

An image of the Activo PC Sénior follows:

Figure 28 - Activo PC Sénior

HP Senior PC's

HP Senior PC's is a combination of hardware with a number of relevant technologies in entertainment and accessibility for seniors [13]. The computers contain applications such as, OnTimeRx, QualiWorld or Claro software. All software and devices are pre-installed and the customer service fulfils the delivery by giving a walkthrough and answering to additional questions. Similarly to the Activo PC Sénior, The available hardware is not only composed by PC's and Laptops but also special keyboards with large keys designed for users with vision problems and "Celery Two-way Printing Mailbox's". This type of printers/fax allows the user to send and receive an email without a computer or internet access by simply using a phone line. The user receives printed emails and image attachments in real-time, just like a fax. It can also send handwritten emails by writing a nickname at the top of a message. The device then converts it to email and sends it to the recipient.

Talking Devices

In the field of assistive technology for seniors talking-devices can provide support for simple tasks such as, knowing the time, checking body temperature, identifying bank notes, etc., through speech synthesis. Most of these applications are directed to the visually impaired, but they can also be used by seniors with difficulty at dealing with technology [14].

Figure 29 – Note Teller, Talking Fever Thermometer and TapMemo

Voice Activated Devices

Voice activated devices like the ones shown in [15] use speech recognition as an interface for tasks such as, controlling television, setting up alarm clock or interacting with an answering machine. As stated in **[19]**, seniors prefer speech as an HCI when compared with other traditional HCIs such as, keyboard. Thus, this kind of devices constitutes an alternative way for seniors to interact with technological devices.

Figure 30 – Voice Interactive Alarm Clock, Voice Activated Remote Control

Caregiving

Caregiving devices described in [16] allow an easier monitoring of the seniors with cognitive disabilities. These kinds of devices allow controlling door locks, locating an individual, remote communication or calling someone just by pressing a button.

Figure 31 – Remote Controlled Doorlock, Channel Wireless Intercom, Portable Color Video & Sound Monitor

Reference Documents for This Chapter

[1] Pew Internet & American Life, "Are Wired Seniors Sitting Ducks?", April 2006

[2] Depression and Social Support Among Older Adult Computer Users, presented August 18 at the 113th Annual Convention of the American Psychological Association

[3] Generations On Line, http://www.generationsonline.com/, last visited on 26-03-2010.

[4] PointerWare, http://www.pointerware.com/c/pages/home, last visited on 26-03-2010.

[5] Eldy, http://www.eldy.eu/, last visited on 26-03-2010.

[6] IBS Diary, http://www.hottimesoftware.com/download-5.htm, last visited on 26-03-2010.

[7] Verbose Text to Speech, http://www.nch.com.au/verbose/index.html, last visited on 26-03-2010.

[8] Baby Sitter and Senior Caregiver, http://www.binaryhouse.com/babysitterandseniorcaregiver.html, last visited on 26-03-2010.

[9] SatTracx, http://www.sattracx.in/, last visited on 26-03-2010.

[10] Nuance, http://www.nuance.co.uk/, last visited on 08-04-2010.

[11] I2Net, http://www.i2net.com.au/Welcome_.html, last visited on 26-03-2010.

[12] OnTimeRx, http://www.ontimerx.com/, last visited on 08-04-2010

[13] HP Senior PC's, http://www.enablemart.com/Catalog/SeniorPC, last visited on 08-04-2010

[14] Talking devices, http://www.enablemart.com/Catalog/Talking-Devices, last visited on 08-04-2010

[15] Voice Activated products, http://assistivetechnologyservices.com/VoiceActivatedProducts.aspx, last visited on 08-04-2010

[16] Caregiving devices, http://assistivetechnologyservices.com/Caregiving.aspx, last visited on 08-04-2010

[17] Emergency medical alert systems, http://assistivetechnologyservices.com/EmergencyMedicalAlertSystems.aspx, last visited on 08-04-2010

[18] QualiWorld, http://qualilife.com/products/index.cfm?id=191&prodType=0&prodTarget=6, last visited on 08-04-2010

[19]Anderson, S., Liberman, N., Bernstein, E., Foster, S., Cate, E., Levin, B., and Hudson, R. 1999. Recognition of elderly speech and voice-driven document retrieval. In *Proceedings of the Acoustics,* Speech, and Signal Processing, 1999. on 1999 IEEE international Conference - Volume 01 (March 15 - 19, 1999). ICASSP. IEEE Computer Society, Washington, DC, 145-148. DOI=http://dx.doi.org/10.1109/ICASSP.1999.758083.