MASTER IN COMPUTER SCIENCE

A Case Study to Enhance the Smartphone's Accessibility for Visually Impaired People

Desart, Simon

Award date:
2016

Awarding institution:
University of Namur

Université de Namur
Faculty of Computer Science
Academic Year 2015–2016

A Case Study to Enhance the Smartphone's Accessibility for Visually Impaired People

Simon Desart

Internship mentor: Carlos Duarte

Supervisor: ______________________ (Signed for Release Approval - Study Rules art. 40)
Bruno Dumas

A thesis submitted in the partial fulfillment of the requirements
for the degree of Master of Computer Science at the University of Namur

Abstract

Nowadays, the use of mobile devices still raises a number of challenges that prevent people with visual impairments from operating their devices efficiently. This thesis aims to provide a global understanding of several concepts as multimodal interfaces but also of the behaviours and concerns of this specific group of users. A new way to interact with a smartphone is proposed by allowing the user to directly access some functionalities through short-cuts. A prototype, inspired by this vision, was developed and was evaluated with the targeted users in order to verify if it can really improve the accessibility of mobile devices for the visually impaired. With this in mind, this thesis also presents a user study where the suitability of the proposed approach was assessed.A performance comparison between the prototype and existing SMS applications was also conducted. The prototype was received positively by the participants. Moreover, it also supported better performance in tasks that involved text editing.

Keywords: *multimodal interaction, accessibility, visually impaired, eye-free interaction.*

Résumé

De nos jours, l'utilisation d'appareils mobiles soulève encore un certain nombre de défis qui empêchent les personnes ayant une déficience visuelle d'exploiter leurs appareils de manière efficace. Cette thèse vise à fournir une compréhension et un aperçu global de plusieurs concepts comme les interfaces multimodales, mais aussi des comportements et des préoccupations de ce groupe spécifique d'utilisateurs. Une nouvelle façon d'interagir avec un smartphone est également proposée, celle-ci permet à l'utilisateur d'accéder directement à certaines fonctionnalités grâce à des raccourcis. Un prototype, inspiré par cette vision, a été développé et a été évalué avec les utilisateurs cibles afin de vérifier si ce mode d'interaction permet réellement d'améliorer l'accessibilité des appareils mobiles pour les malvoyants. Dans cette optique, cette thèse présente également une étude menée auprès des utilisateurs cibles dont le but était d'évaluer la pertinence de l'approche proposée. De plus, une comparaison des performances entre le prototype et une application de SMS existante a été menée. Le prototype a également été reçu positivement par les participants. En outre, il a aussi montré une meilleure performance dans des tâches impliquant de l'édition de texte.

Mots-clés: *interaction multimodale, accessibilité, malvoyant, interaction sans contact visuel.*

Acknowledgements

The work described in this thesis is in the continuity of a three-and-a-half-month internship at the University of Lisbon in Portugal and more particularly in the LaSIGE (**La**boratório de **S**istemas **I**nformáticos de **G**rande **E**scala) research unit of the Computer Science department of the Faculty of Sciences. This internship is part of the master's program in computer science of the University of Namur. It was proposed by Pr. Bruno Dumas and was supervised on-site by Pr. Carlos Duarte.

So first, I would like to thank Pr.Dumas for his availability and his advice during the writing of this thesis but also before during the internship.

For his great welcome, I wish to express my sincere thanks to Pr. Carlos Duarte. I also want to thank the other members of the office and also David Costa for his help during the user studies.

Finally, I thank my family for their support and encouragement through my academic curriculum and during this thesis' writing.

Contents

List of Figures

List of Tables

Introduction

Despite important improvements in mobile device's accessibility, the use of those devices still raises a number of challenges that prevent people with visual impairments from operating their devices efficiently. According to the World Health Organization [34], it is estimated that in the industrialized countries, one person out of thousand is blind and one out of hundred is considered as visually impaired.

These problems especially occurred in uncontrolled environments, like when the visually impaired is walking outside in the street, because both their hands are not free and the unfamiliar surroundings. This thesis aims to gather information about the visually impaired and then, as mentioned in the title, to evaluate a possible way of interaction through a case study in order to see if it is conclusive. The strength of the proposed prototype resides in the multimodal interaction. Indeed, multimodal interfaces can help to enhance the accessibility for this special group of users by offering them alternative and a more human way to interact with those devices.

Therefore, the concepts behind these specific interfaces are presented in the first chapter of this thesis. Other concepts as a modality, context adaptation or accessibility are also introduced in this chapter.

Understanding the visually impaired behaviour is also mandatory to assure designing systems according to their specific needs. The second chapter first expands some key concerns of people with visual impairment when it comes to using mobile devices or in general as the safety and privacy, the independence and autonomy as well as the conformity and the social acceptance. Then, a list of possible modalities that fits their disabilities is compiled. This list aims to be diversified as possible by presenting auditive, tactile, haptic and vestibular modalities. Theses modalities are also compared regarding criteria as the expressiveness, the private life's respect, the affordability or the ease of use. Finally, some interesting technologies are mentioned at the end of the chapter.

Chapter 3 focuses on the output side of multimodal interfaces, more especially on the fission and explores the current use of this term. This chapter concludes the first part of this thesis, which consists mostly in gathering, summarizing and comparing information from scientific papers. The second part is more focused on proposing a more accessible way to interact with mobile devices for the visually impaired and also to evaluate the proposal with the targeted users.

Indeed, chapter 4 introduces a conceptual architecture to enhance the accessibility of smartphones. The main idea is to directly access the desire functionality through short-cuts and so to reduce the time that a user spends navigating the user interface to reach this functionality. The accessibility services, offered by the Android Accessibility API, are also explained. The functioning of one of these services developed to retrieve the components of the smartphone's interfaces is detailed as well as its limits and the problem encountered during its development.

In the continuity of chapter 4, chapter 5 introduces a prototype based on the main idea of using short-cuts presented earlier. This prototype of an SMS application let the user writes, edits and finds a contact through a combination of gesture and speech. The functionalities implemented are described as well as the rationale behind some design decisions. User studies were performed to assess the performance of the prototype in comparison with a standard application coupled with a standard accessibility service. The user study protocol is explained and then, the results with sighted users followed by the ones with unsighted users are detailed. The results gathered during the evaluations with unsighted users were the basis for a scientific paper which can be read in the appendix B on page 97. This paper was submitted in May 2016 to the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2016).

To conclude, a brief summary of what is presented in this thesis is made. Then, a critical outlook over the developed prototype is given with its limits but also the perspectives for some possible improvements and future works as well.

Chapter 1

General Concepts

This chapter aims to briefly present some concepts mandatory to fully understand this thesis, such as multimodal interface, context adaptation or the concept of accessibility.

Contents

1.1 Modality

One possible definition by Nigay [32] is the next one : *"A modality m is defined as the couple* $m = < r, d >$ *where r denotes a representational system, and d the physical I/O device used to convey expressions of the representational system."*

So a modality can be seen as a path of communication between the human and the device, it is the information structure as it is perceived by the user. Human communications are consequently inherently multimodal and to each human sense corresponds a modality : visual for the sight, auditive for the hearing, tactile for the touch, olfactory for the smell, gustatory for the taste and vestibular for the balance. A modality has other properties: it can be an input or an output, atomic or continuous, raw or interpreted and active or passive. The two most common modalities used by humans are speech and gesture. Theses modalities are both input and output and can, of course, be combined.

It is also possible for a modality to be decomposed in different modalities regarding the level of abstraction. For instance, each characteristic of the sound (pitch, rate, volume,...) can be used as a new modality.

1.2 Multimodal Interface

A multimodal interface, as its name indicates, will use several modalities. This section will briefly explain what a multimodal interface is.

Multimodal interfaces are by nature a good option to make adaptive interfaces by switching from a modality to another. They also allow the exploitation of the communicative and perceptual capabilities of the user by offering more natural interaction ways using all the five senses.

Other advantages of multimodal interfaces are that they will enforce the reliability and the robustness by combining different sources of information as well as the flexibility thanks to the multiple modalities available as mentioned by Dumas *et al.* [19].

On the other hand, multimodal interfaces are generally time sensitive and use a large amount of resources to work correctly.

Multimodal interfaces rely on two main concepts: the fusion and the fission. The fusion is applied on the input side to extract information from the different input modalities and can be applied on different levels (data, feature or decision). At the output side, the fission consists to choose the best way to render a message using all the modalities at disposition regarding the context, this will be explored in the chapter 3 on page 39. The figure 1.1 here below illustrates these concepts and presents the interaction loop with the user.

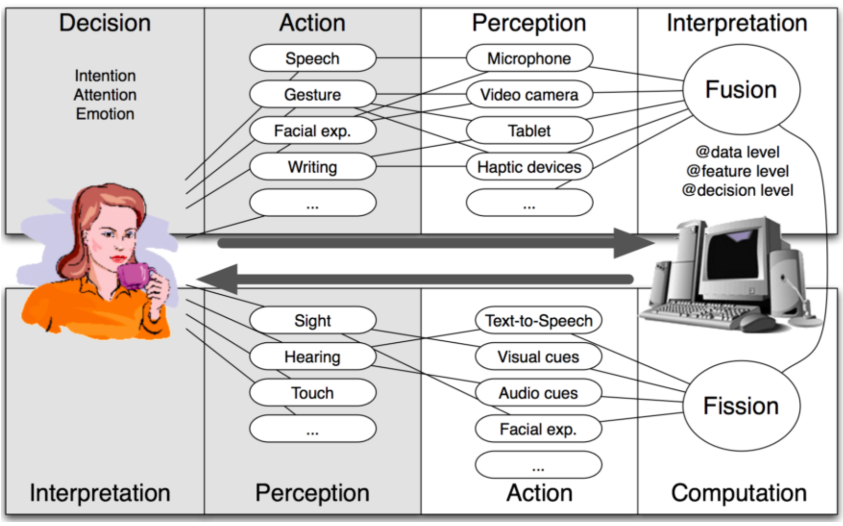

Figure 1.1: A representation of multimodal man machine interaction loop [19]

It is also important to mention that multimodal interfaces can be used to foster the memorization by distributing the information along complementary modalities, which is confirmed by a survey led by Appert and Dubois [3], or by making the information redundant with the use of several modalities to convey the same information.

It can also be noticed that a multimodal interface is inherently multimedia but this relation is not reflexive. Indeed, a *media* is a support for communication, it represents formats for presenting information. For instance, a mail application allows to convey different types of media (text, image, video,...) but is usually not multimodal.

1.2.1 The CASE Model

Nigay *et al.* [33] propose a design space to classify multimodal interactions, this model is defined along three dimensions:

- ***Levels of Abstraction***: regarding the inputs, the data received from a device can be processed at several levels of abstraction and for the outputs, the data may be produced from abstract symbolic data or from a lower level of abstraction with less meaning. For instance, speech input can be stored as a signal, as sequence of phonemes or as meaningful parsed sentence. On the other hand, a vocal message can be produced from an abstract representation, from a text or simply by playing a recorded message.

- ***Use of modalities***: this axe focuses on the composition of the modalities Two possibilities are presented: *sequential* or *parallel*. So, multiple modalities can be employed simultaneously or one after another sequentially.

- ***Fusion***: it covers the possible combination of different types of data, it is the relation between the information gathered produced through the modalities. Two possible values are given for this relation: *independent* or *combined*.

Put together these three dimensions lead to the following classification:

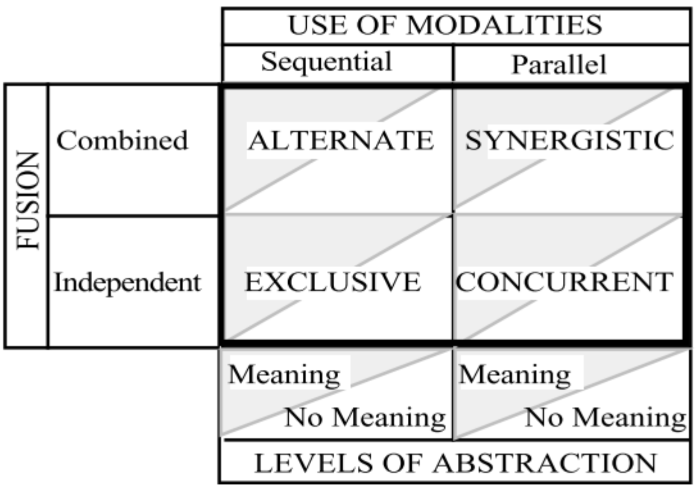

Figure 1.2: The multi-feature design space [33]

Therefore, it can be concluded that this classification space focuses only on the characteristics of the system which are the concurrency of data processing and data fusion to describe both input and output interface of multimodal systems.

1.2.2 The CARE Properties

Coutaz *et al.* [16] propose four properties to easily characterize and assess aspects of multimodal interaction:

- **C**omplementarity
- **A**ssignment
- **R**edundancy
- **E**quivalence

Complementarity characterizes a system where several modalities have to be used to reach a certain state within a temporal window, and this state cannot be reached otherwise with a single modality taken individually.

Assignment indicates a system where only one modality can be used to reach a certain state, it expresses the absence of choice. Either there is no choice at all or even if there is a choice, the same modality will always be opted for.

Redundancy defines a system where multiple modalities have the same expressive power and if all of them are used within the same temporal window, the state of the system does not change. It is a repetitive behaviour that does not increase the expressive power of the system.

Equivalence describes a set of modalities when for reaching a certain state, it is necessary and sufficient to use any one of them. It expresses the availability of choices between these modalities without imposing any temporal constraint on them.

These properties cover both input and output modalities and so can be applied to the design and implementation of both of them and, of course, they are not exclusive from each other.

1.3 Context Adaptation

Nowadays, it seems that the terms context-sensitive and context-aware interfaces are widespread. Indeed, the need for interfaces that are aware of their context and that are able to adapt themselves is more actual thanks to the hegemony of smartphones and the emergence of the Internet of Things paradigm. Doukas and Antonelli [18] defined these concepts as follows: *"Context awareness refers to the capability of the computing or networking applications to be aware of the existence and characteristics of the user's activities and environments. A system is context-aware if it can extract, interpret and use context information* *and adapt its functionality to the current context of use."*

Hence, the context is a central concept. As defined by Calvary *et al.* [11], the context of use regroups theses 3 classes of entities:

1. The ***user*** of the system who can be described according to his cognitive abilities and his physical capabilities;
2. The ***physical environment*** where the interaction with the system occurred. It's defined by Coutaz and Ray [17] as *"the set of objects, persons and events that are peripheral to the current activity but that may have an impact on the system and/or user's behaviour, either now or in the future"*;
3. The ***software*** and ***hardware components*** that are used for interacting with the system. They can be expressed in terms of resources as memory size, screen size, complexity, network consumption, ...

The adaptation may be led by these 3 classes. The classification of adaptation made by Thevenin *et al.* [43] goes in that way. It is stated that the adaptation can be made according to 4 axes as shown on the figure 1.3:

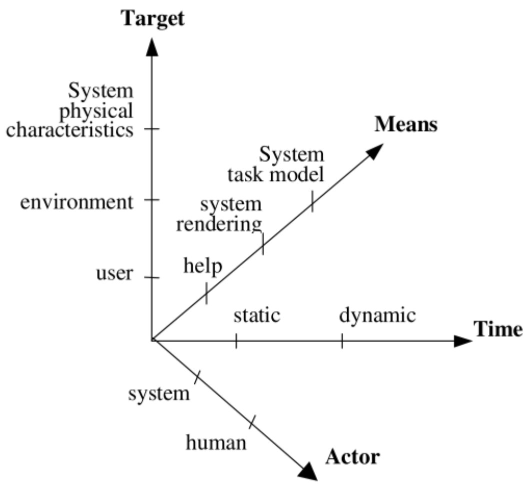

Figure 1.3: A design space for adaptation [43]

1. The ***time***: regarding if the adaptation is dynamic (occurs at runtime) and/or static (occurs between two sessions);
2. The ***target for adaptation***: for whom or what the adaptation is intended for. It corresponds approximatively to the 3 classes of entities mentioned above;
3. The ***means of adaptation***: it refers to the software components involved for performing of the adaptation;
4. The ***automaticity***: it is also important to mention that the adaptation can occur automatically or in response to a user's request. It refers to

the two complementary properties of adaptation: *adaptability* and *adaptivity*. A system is adaptable if it lets the user control some parameter to customize it. On the other hand, a system is adaptive if it is capable of adapting itself automatically without any user's interaction.

1.4 Accessibility

For the W3C [13], *accessibility* is defined as followed : *"Accessibility addresses discriminatory aspects related to equivalent user experience for people with disabilities, including people with age-related impairments."*

It's closely related to *usability* which is also defined by the W3C as : *"Usability and user experience design is about designing products to be effective, efficient, and satisfying. Specifically, ISO defines usability as the extent to which a product can be used by specified users to achieve specified goals effectively, efficiently and with satisfaction in a specified context of use (in ISO 9241-11). "*

Regarding these definitions, even if accessibility focuses on people with disabilities, an improvement of the accessibility will also probably improve usability for everyone. Accessibility standards also have an important role in accessible design. Indeed, using and understanding these standards as well as the basic accessibility principles help to develop directly accessible prototype but also allows users to provide useful and relevant feedback during evaluation.

Chapter 2

Visual Impairments: Characteristics and Challenges

According to the World Health Organization [34], 285 million people are visually impaired worldwide. Within these 285 million, 39 are blind and 246 have low vision. A possible definition for visual impairment is the following given by the *International statistical classification of diseases, injuries and causes of death*, 10th revision (ICD-10):

- *Visual impairment includes low vision as well as blindness;*
- *Low vision is defined as visual acuity of less than 6/18, but equal to or better than 3/60, or a corresponding visual field loss to less than 20 degrees in the better eye with best possible correction (ICD-10 visual impairment categories 1 and 2);*
- *Blindness is defined as visual acuity of less than 3/60, or a corresponding visual field loss to less than 10 degrees in the better eye with best possible correction (ICD-10 visual impairment categories 3, 4 and 5).*

This chapter first presents some particular concerns specific to the visually impaired. Then, a list of possible modalities that are used or could be used by them is compiled and then compared. Finally, some interesting technologies are mentioned.

Contents

2.1 Special Concerns of Visually Impaired People

To be able to produce suitable designs for the visually impaired, it is important to understand their needs, their behaviours as well as their concerns in some particular contexts. For instance, blind people cannot drive and so they are more likely to use public transport to ensure their mobility. When using public transport, a survey led by Azenkot *et al.* [5] shows that blind people give importance to two key values: safety and independence. This section aims to describe some specific concerns as those listed just above.

2.1.1 Safety & Privacy

Blind people can be reluctant to use technologies that expose personal information as it is often the case when using speech recognition or text-to-speech engine. Some blind users say that they feel safer using Braille because people cannot see what they are doing. They also do not want to lose attention because of intrusive notifications or feedback. It can also be challenging for visually impaired people to preserve their privacy in their daily lives. Indeed, on the contrary of sighted people, it is difficult for them to be able to monitor what surrounds them and so to protect themselves from privacy threats.

Wu and Adamic [44] analysed the use of social media (especially Facebook) by visually impaired people and findings showed that they shared photos as often as an average user. But on the contrary of a sighted user, a visually impaired one cannot check that the photograph does not include embarrassing or private content. This proves that privacy needs to be taken care of in different settings: physical, online and at the junction between the two when they are using technology in public. Indeed, regarding online privacy, the visually impaired found online banking and shopping more accessible than visiting actual stores but they stipulated that it was not a perfect solution because of concerns about the transaction security.

According to Ahmed *et al.* [1], most users are not aware of mobile security threats and nearly none had enabled locks on their phone. Along these threats, eavesdropping and shoulder surfing are the most unknown ones which are, of course, reinforced by the fact that they do not know if other people are nearby. To facilitate password management, some recorded their passwords in a file and then used screen readers to retrieve them, which put them at risk for eavesdropping.

2.1.2 Independence & Autonomy

The visually impaired people often asked other people for information about the environment that surrounds them but it is not always possible to find someone and there is no guarantee that the information is reliable. So they prefer having more information that enables them to be more independent and prepared, giving them an easy access to this information makes even more sense. It counts not only for gathering information but also for other tasks as making a call, one of a participant in a survey led by Azenkot *et al.* [5] said : *"Communication is very important. I want to get confirmation that I was understood. It's hard to trust another person to make a call for you... I prefer to be as independent as possible."* It is highly coupled with the privacy concern since they potentially have to reveal private information. For instance, to fill a medical form or the pay a bill.

Mobile phones have given the possibility to visually impaired people to increase their freedom to act more independently while remaining in contact with help if it is required. But it can also create a dependence on theses technologies and so to avoid being trapped if the device failed, they have a tendency to carry a back-up device. For instance, another GPS in case of the one embedded in the mobile phone crashed.

Brady *et al.* [8] used the term *friendsourcing* to indicate the use of social networks to ask visual questions at their friends but they have a tendency to hesitate to do that because they do not want to bother their networks and desire to appear independent and not to give the impression of being helpless.

2.1.3 Conformity & Social Acceptance

Blind people usually do not want to be treated specially because of their disability. And so they could be reluctant to use technologies that can make them look weird. For instance, they seem to be more disposed to use a wearable device if they know that it will also be used by common people. It is the same for a gesture-based system where they will prefer to use more discreet gesture if possible to avoid being noticed or with noisy sound based notifications that they will probably turn off.

Moreover, Kane *et al.* [25] noticed in 2009 that people with visual impairment often use devices designed for the general population in combination with specialized devices to overcome the limited functionalities or the price of the latter.

A study led by Shinohara and Wobbrock [40] confirmed that disabled people were more likely to abandon an assistive device if they did not accept their disability, if the device socially excluded them (made them feel different from their peers), or if the device significantly clashed with cultural values. Disabled people acknowledged too that aesthetic and design issues distinguished their devices from mainstream devices with similar functionality. It was also found that they were also aware of misperceptions that others have about assistive technologies, two common misperceptions were particularly mentioned:

- *Assistive technologies functionally eliminate a disability*; people have a tendency to think that with these devices, disabled people can be normal and that it erases totally their disabilities. A quote of a participant showed that well: *"I have a friend and we were talking about being disabled. She said, 'well, you have your VoiceNote.' It's like people think you can be normal... because you have some technology... you're still not visually normal. But... I don't know, she had the misperception that, well, I could just live a normal life because I had a VoiceNote... Well, I can live better. But I don't... see, I'm comparing it to a visual life. And, I thought, I still don't live a visual life, even though I have assistive technology. But, I know the difference between sighted life and not sighted life."*

- The other one misperception is kind contradictory to the first one, *people with disabilities that used assistive technologies are helpless without them.* This misperception coupled with the fact that such devices usually reinforced the visibility of the disability could cause a disabled person not to use an assistive device. Indeed, some participants said that they did not want to use white canes and hearing aids because the social stigma associated with these identified them as less capable.

Moreover, it did not matter if the participants were totally blind, had low vision, were born blind or became blind later in life, they were all aware of the social stigma. Therefore, it makes sense that the participants did not want to be associated with these perceptions.

2.1.4 Training & Changing Abilities

To overcome the accessibility gaps, visually impaired people need to learn how the system works or to configure it and then possibly train to be more efficient. So they practice at their home in order to memorize the manipulation to perform a certain task to be able to accomplish it more quickly if necessary.

It is also possible that based on factors like the stress or the fatigue, their visual abilities or their attention decreased over time.

2.2 Possible Modalities for People with Visual Impairment

Blind people cannot rely on the most common used modality, the vision, which is usually essential to assure an effective usability. This section presents a list of possible modalities which can be used as input or output for blind people.

2.2.1 Auditive Modalities

The main challenge with these auditive modalities is not to overwhelm the user with too much additional auditive information. Indeed, people with visual deficiency rely on environmental audio cues to correctly perform certain tasks. Moreover, if sounds are used to give feedback, they have to be non-intrusive otherwise the user will shut them down.

2.2.1.1 Speech

Text-to-speech service and speech recognition can be used respectively as output and input.

Advantages : The main advantage of a text-to-speech service is its high expressiveness and so, it allows to give clear feedback. Regarding the speech recognition as input, it enhances the interactivity, is relatively fast and it is naturally portable.

Drawbacks : A special attention must be taken when using this modality to ensure the safety and the privacy. Indeed, an interview led by Azenkot *et al.* [5] showed that when they were on the go, blind people were not able to use comfortably speech technology. Difficulty to hear, a lack of privacy (disclosure of information in public spaces) and the safety (loss of some audio cues) were the main problems stated by the interrogated blind people.

Regarding the speech recognition, another study of Azenkot *et al.* [4] showed that it can be 5 times faster than the tactile keyboard. Indeed, with speech input, participants entered text at a rate of 19.5 words per minute, while they entered only 4.3 words per minute with the keyboard coupled with iOS's VoiceOver. However, it also pointed out that editing the errors of the speech recognition can be frustrating and that the user spent 80 % of his time editing and so, it is perhaps more suited for short messages. Moreover, speech recognition requires an Internet access to be efficient. Indeed, the performance of offline speech recognizers are quite poor due to the fact that they are based on the recognition of the sound of each phonemes. On the other hand, online recognizer compared the audio sample directly to a huge sound database thanks to big data algorithms and this method is far more efficient.

2.2.1.2 Earcons [29]

The principle is to use non-natural sounds based on a symbolic and arbitrary representation to provide information to the user about some operation or interaction. Well-known earcons are the ones of Windows for starting up or shutting down.

Advantages : The main advantage of earcons is that they can be easily parametrized.

Drawbacks : A training phase is usually required to let the user know the signification of the sound and an earcon must be easy to understand and remember.

2.2.1.3 Auditory Icons [9]

Auditory icons rely on every-day-life sounds and so it is based on analogical representation. A famous auditory icon is the one when the user empties the Windows Recycle Bin.

Advantages : The main advantage is this analogical representation. Indeed, the user already know the meaning of the sound and so a training phase is not mandatory.

Drawbacks : On the other hand, auditory icons cannot always be parametrized and it is not always possible to find a natural sound that matches what it is intended to represent to ensure that the analogy works.

2.2.2 Tactile Modalities

2.2.2.1 Braille

It is possible to combine mobile applications with a Braille note taker. Indeed, Azenkot *et al.* [5] developed a framework called *MoBraille* which enables to develop applications that interface with Braille note takers without caring about proprietary or device-specific knowledge through WiFi. Developing these applications consists in writing an HTML page and a Java servlet that runs on Android. Thanks to that framework, Braille note takers can exploit the functionalities of an Android phone as the GPS, the compass or the 3G network connectivity. This is made possible by using HTTP requests over a WiFi connection as shown in the figure 2.1 here below:

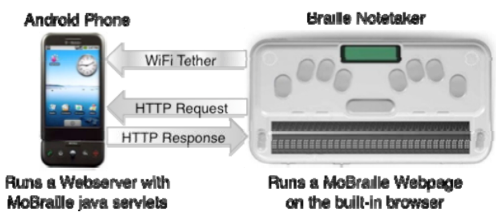

Figure 2.1: The MoBraille Framework [5]

Frey *et al.* [21] developed *BrailleTouch*, an application to entry text through Braille for mobile devices. This application is based on the Braille binary matrix which is a 3 by 2 matrix that encodes up to 63 characters (the all-null state is not used). The figure 2.2 presents how the alphabet is encoded through the matrix where patterns can be found and which can be used as mnemonics. For instance, a simple pattern used is that the letters A through J only used the top four positions.

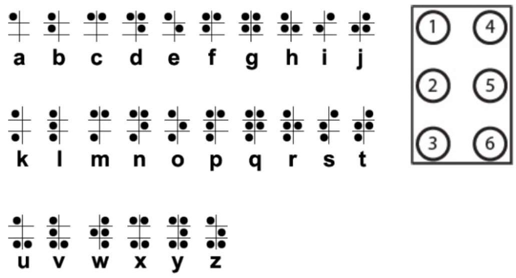

Figure 2.2: The Braille Alphabet

BrailleTouch reused this structure (as shown in the figure 2.3 on the right), the authors said that the key aspect of the technology is that it has fewer buttons than fingers and thus, the user does not have to move fingers around to find the right sequences and combinations to type. The application can also be used as a pedagogical tool to learn Braille through the possible oral feedback.

Figure 2.3: BrailleTouchs input surface faces away from the user [21]

Advantages : It is less disruptive than speech and so ensures safety but also privacy while allowing to convey a lot of information. The information displayed is also better retain than with a text-to-speech based system. When evaluating BrailleTouch, Southern *et al.* [41] found out that six expert Braille typist averaged speeds was 23.2 words per minute with 14.5 % of errors which is way better than 4.3 words per minute with iOS's navigation system, VoiceOver [4].

Drawbacks : However, theses specialized devices can be very expensive and too dependent on the manufacturer as stated by Kane [25] and have usually limited functionality. The speed of use can depend on the mastery of the language of each individual. It could also be problematic to use a Braille note taker following the weather like the rain or the cold if the user wants to wear gloves. Moreover, the user has to carry one additional item which has its own expense, batteries and setup time.

2.2.2.2 Touch Screen

Touch screens have become more mainstream and frequent over the past ten years and are present in a lot of everyday technologies as mobile phones, tablets or public kiosks. The range of gestures than can be performed on a touch screen is quite diversified: a tap on a special location of the screen, directional swiping gestures, multi-touch gestures, gestures that represents geometric shapes or finally, gestures based on symbols as letters or numbers.

Regarding the use of touch screens, it appears that the behaviour of a blind person can differ from the one of a sighted person. It is possible that the blind user will prefer different gestures or that they will perform the same gesture differently than common people. Indeed, even a sighted person tends to perform gestures differently when they are in a situation with low visual feedback. A survey led by Kane *et al.* [27] presented some interesting conclusions:

- When they were asked to create gestures, they often used non-conventional multi-touch gestures in which the participant held one finger down on a specific area of the screen while performing the gesture with a second finger or hand. The use of a virtual mode key was probably motivated by the one present on computer physical keyboards and to reduce potential conflicts between gestures. On the opposite, no sighted participants created gestures based on a physical keyboard layout while 21.6% of the gestures invented by blind participants during the study were based on this layout.

- Blind people tended to represent bigger gestures than sighted people and the variation in size between two representations of the same gesture was also bigger. It was also noticed that blind people took on average twice the time of a sighted person to perform a gesture and so, in consequence, speed should not be used as a recognition feature. Concerning shape gestures, it is also likely that the form will not be well closed as shown with the circle in the figure 2.4, so that the beginning point will not coincide with the ending point. The last element on the figure 2.4 showed that the lines of some blind participant seemed to be less steady. Theses results may be important to make an efficient gesture recognizer and be sure that gestures executed by blind people will be recognized correctly.

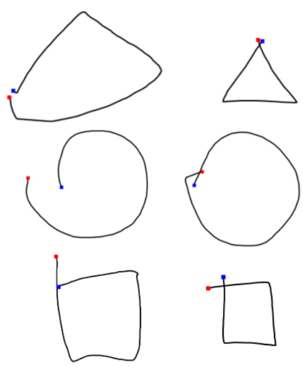

Figure 2.4: Gesture of different shapes produced by a blind person (left) and a sighted person (right) [27]

- Locating specific points on the screen is inherently complicated for a blind user, so using the corner and the edge can be useful to help them especially by placing critical functions in theses areas. Another solution is also to reduce the demand for location accuracy by increasing the size of the area to locate or to offer the possibility at the user to know what is near on the screen after pointing a random point.

- Blind participants have difficulties with symbolic gestures because a lot of them are unfamiliar with them and so they hardly ever use this kind of gesture when they are asked to invent a gesture for a specific task. Moreover, some of them were not able to perform correctly this type of gesture. Thus, avoiding symbols used in print writing could be a good guideline to guarantee a better experience of use for a blind user.

Advantages : The main advantage of touch screens lies in the fact that they are very mainstream and so visually impaired people are more and more in contact with them which means that they are more likely to seize this technology. Moreover, studies as the one lead by Kane *et al.* [27] have shown that early-blind and late-blind people have higher tactile sensitivity in their fingers than sighted people.

Drawbacks : Designers usually do not know how blind people actually use touch screens and they are not taken into account in the development. The aforementioned conclusions can also be obstacles to an efficient use of touch screens by non-sighted people.

2.2.3 Haptic Modalities

Haptic feedback consists mainly in using the sense of touch of the user to convey information. For instance, it could be vibration but also the force feedback of a joystick. Regarding the vibration, it is different than merely vibrate as an alarm, haptic feedback uses advanced vibration patterns to convey information and is meant to enhance the user experience.

2.2.3.1 Vibration

Quick feedback can be given through vibrations without disturbing the user too much. Indeed, for instance, a navigation interface can rely on vibration to guide the user. Pielot *et al.* [35] developed PocketNavigator, a demonstrator that adds tactile feedback to an Android navigation system. It basically acts like a compass that points at the next waypoint with its distance and its direction encoded in vibration patterns. Unlike traditional navigation systems that are based on turn-by-turn instruction, the developed system gives continuous feedback through vibration cues and, moreover, the phone can stay in the user's pocket.

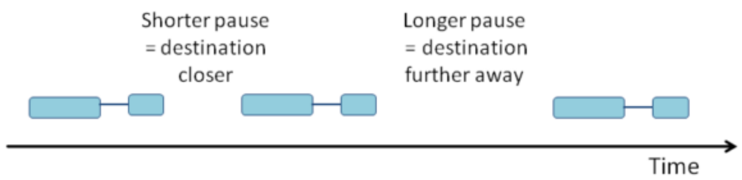

Figure 2.5: Representation of the distance by the PocketNavigator [35]

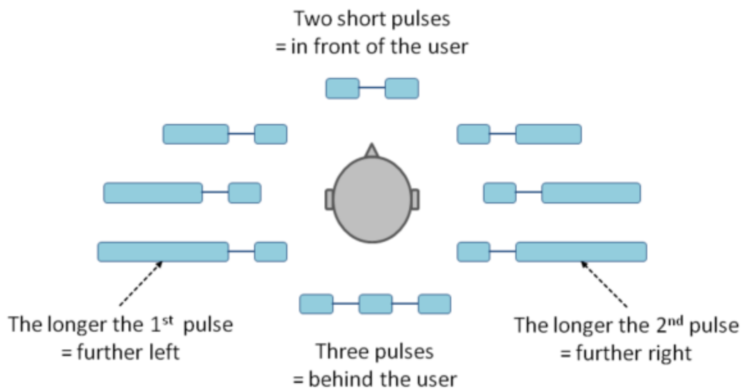

Figure 2.6: Representation of the direction by the PoketNavigator [35]

This feedback is only based on two pulses which duration varies according to the direction as well as the distance as shown on the figures 2.5 and 2.6.

Advantages : This technology is rather not intrusive. Plus, the emergence of smartwatch which can vibrate remotely thanks to Bluetooth may also help to use more vibration without the needs of intrusive devices as stated by Appert *et al.* [2].

Drawbacks : The possible feedbacks are quite limited. Indeed, typical vibration motors in smartphone can only be turned on and off. So basically, it is only possible to parametrize the duration and the rhythm but not the amplitude. For instance, for the PocketNavigator, the distance and the direction are modelled as shown in the figures 2.5 and 2.6. These ones probably require a training phase to guarantee the comprehension of the feedbacks.

2.2.4 Vestibular Modalities

2.2.4.1 In-air Gesture

Gestures are interesting because they are an integral part of the language and is closely coupled with the speech. So the idea is to communicate with the technology as we do naturally with other people. They also allow to manipulate virtual objects as we manipulate physical objects. So the possibilities are endless.

One of the first systems to use gestures was the famous *Put-That-There* made by Bolt in 1980 [6] where gestures and speech were complementary used to respectively point a location and give the system a command. A typical command was : *"Put a magenta square there"*.

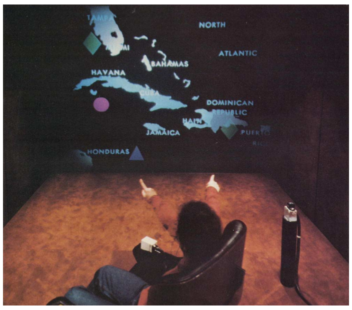

Figure 2.7: The *Put-That-There* system operating [6]

Costa *et al.* [15] used surface electromyography signals (sEMG) to recognize arms and forearms movements. This especially enhanced the usability for blind people when they are on the go. To capture these signals, they used sEMG sensors which consists to place three electrodes on the forearm as shown in the figure 2.8 on the right.

Figure 2.8: Placement of the sEMG electrodes [15]

A recent breakthrough, led by Google's Advanced Technology and Projects group (ATAP), could be a game changer. Indeed, the project ***Soli*** [42] is a chip that incorporates the entire sensor and antenna array into an ultra-compact 8mm x 10mm package. It is described as : *"Soli is a new sensing technology that uses miniature radar to detect touchless gesture interactions. Your hands are the only interface you'll need."* Their goal is to create a ubiquitous gesture interaction language that will allow people to control devices with a simple, universal set of gestures. These gestures are designed to mimic familiar interactions with physical tools, using this metaphor is an easy way to communicate, learn and remember the Soli interactions. For instance, the following gestures (shown on the figure 2.9) are proposed:

"Imagine an invisible button between your thumb and index fingers you can press it by tapping your fingers together. Or a Virtual Dial that you turn by rubbing thumb against the index finger. Imagine grabbing and pulling a Virtual Slider in thin air."

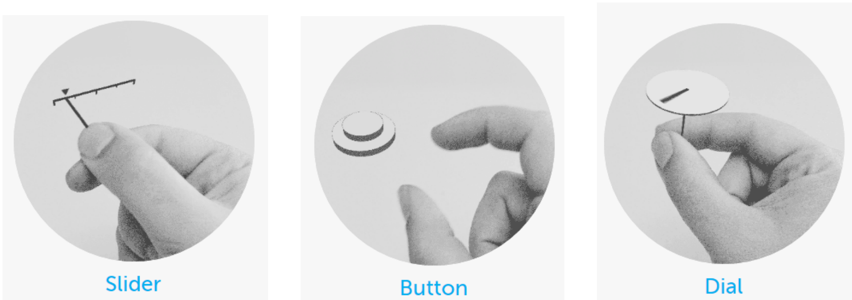

Figure 2.9: Soli Virtual Tool Gestures [42]

Regarding feedback, they say that it is generated by the haptic sensation of fingers touching each other and so, the interaction feel physical and responsive. Finally, the big advantage is that it can be embedded in all kinds of devices as wearable, phones, computers, cars or IoT devices. Soli has no moving parts, it fits onto a chip and consumes little energy. Moreover, it is not affected by light conditions and it works through most materials. Therefore, the potential is huge and offers a lot of possibilities.

Advantages : It allows to interact with a system without requiring any physical interaction, and so can provide a quick access. For instance, if we take a smartphone, it is quicker to make a gesture to launch an application than retrieving the phone from the pocket, unlocking it and then launching the application.

Drawbacks : The user needs to know where they have to perform the gesture to ensure that it can be sensed and recognized; it can even be more problematic if multiple systems are present, the user also has to know how to reach the wanted system and not have unintended effects on the others. And so, it can be critical to give users fine feedback. Indeed, most of actual systems using gesture do not give any feedback and users only know if the action was performed if they can perceive the effect of their action. Such system needs to give sufficient information about the interaction mechanisms to be effective. Regarding social acceptability, Rico and Brewster [36] findings show that there is a significant relation between audience and location with the willingness to perform certain gestures. Therefore, this must be taken into account when creating such systems. It was also found that users are fonder of subtle movements, movements that are similar to what already exists in current technology (for instance, gestures similar to the ones used with a touch screen), and movements similar to the ones used in our everyday lives (like feet tapping, shaking or rhythm gestures) or enjoyable movements (gestures that provided a high amount of satisfaction, whether from the actual feeling or appearance of the gesture itself). On the other hand, participants in these studies stated that uncommon, large or noticeable movements would look weird in public settings. Physically uncomfortable gestures or gestures that interfere with communication like head nodding were also not appreciated by the participant.

2.2.5 Comparison Between the Proposed Modalities

The different modalities presented will be approximately compared along four criteria:

- The ***expressiveness*** of the modality: Is-it possible to convey rich information through it? (The more + are marked the more the modality is considered expressive);
- The ***intrusion***: How intrusive and discrete is the modality and how it can affect the privacy?(The more + are marked the more the modality preserves the privacy);
- The ***ease of access***: How easy is to learn how to use the modality and is-it affordable? (The more + are marked the more the modality is easy to use or affordable);
- The ***visibility*** of the modality: Does the use of this modality reinforce the visibility of the user's handicap? (The more + are marked the more the modality reinforces the handicap).

Table 2.1: Global comparison between the presented modalities

Modality *(Input/Ouput)*	***Expressiveness***	***Intrusion***	***Ease of Access***	***Visibility***
		Auditive Modalities		
Speech (I/O)	+++	+++	+++	++
Earcons (O)	+	++	+	+
Auditory Icons (O)	++	++	++	+
		Tactile Modalities		
Braille (I/O)	+++	+	+	++
Touch Screen(I)	++	+	+	+
		Haptic Modality		
Vibration (O)	+	+++	++	+
		Vestibular Modality		
In-air Gesture (I)	++	+	++	+

2.3 Interesting Technologies for Visually Impaired People

2.3.1 Access Overlays

Kane *et al.* [26] defines access overlays as follows: *"Access overlays are accessible interaction techniques that improve touch screen usability while preserving the original spatial layout of an application."*

Basically, it will gather information about the components and the layout of the screen and then lets the user interact with them in a more convenient way regarding his disabilities.

The access overlays presented in this article aim to be used with large touch screens where the search time to find a specific target on a screen is obviously longer than with small screens. One big advantage of access overlays is that they do not rely upon hardware modification such as adding a physical button or placing a physical overlay atop the screen. Access overlays are software-based and so, they do not require any modifications to the underlying touch screen hardware.

One common technique used by access overlay is to convert the interface into a linear list of targets, this process is called *linearisation*. That allows blind users to quickly know what are the on-screen items without having to search the entire screen. Nevertheless, this technique alone does not preserve the original layout of the screen. However, it could be for example coupled with the *edge projection overlay*, as shown in the figure 2.10, which provides the benefits of linearisation while preserving the original spatial layout of the screen.

Figure 2.10: Edge projection of the items resulting of the linearisation [26]

2.3.2 Wearable Devices

It is also possible for people with visual impairments to use wearable devices as stated by Ye *et al.*[45] to improve their ability to access information and to interact with devices. Indeed, the advantages offer by wearable devices are multiple: a quick and convenient access, it is safer, their size is smaller and it allows a more discreet and private use when in the same time supporting the use of the phone on the go. Moreover, wearable devices are becoming more common as the personal tracking wristband or more sophisticated smart watches. However, it is crucial that it must not be invasive and the aesthetic criterion has to be taken into account as well.

2.3.3 Mobile Phones

The evolution of phones was beneficial for the visually impaired people making them more independent but it clearly does not mean that they can use them in an easy and effective way. Indeed, with first the appearance of display screens and then the emergence of touch screens in correlation with the disappearance of physical buttons which were becoming too tiny, they are not at their advantage at all.

However accessibility options have also emerged with options such as text-to-speech or the possibility to adjust the size of the font as well as the contrast of the screen. Some participants of a survey led by Kane *et al.* [25] even stated that, rather than becoming more accessible, their newer phones were less accessible than previous they had possessed.

According to the same survey, some situations can also have a negative effect on their facility to use a mobile phone:

- ***Crowded space***: some participants had difficulties when using mobile devices on the street or on a bus;
- ***Lighting and weather***: some participants with low vision mentioned that their screen was not readable in very bright or dim light and, of course, the rain or the cold (through the use of gloves) can also be problematic;
- ***Use while walking***: this can be challenging by reducing the motor control or the situational awareness of the user which will only use a mobile device while walking in a familiar area. On the other hand, if the person stop walking to use his phone, it could cause other difficulties as blocking the sidewalk or standing in a critical area.

2.3.4 Screen-Reading Software

This section presents briefly the two most used screen reader for mobile phones which are Google's TalkBack [1] and Apple's VoiceOver [2].

2.3.4.1 TalkBack

TalkBack furnishes two ways to explore the screen. The first one is the ***basic touch exploration***, the user just needs to slowly drag one finger around the screen and TalkBack will announce the different components as the user drags his finger over them. The second one is the ***linear navigation***, which is similar to the linearisation presented in section 2.3.1, the user swipe left or right to explore one item at a time. In both cases, to select the focused component, the user has to double-tap anywhere on the screen.

Regarding text input, to type a letter, the user slides his finger over the keyboard until he hears the character he would like to type and then, he lifts his finger to actually type the focused key. This only works with the default Android keyboard. To review this text, the user has to press the volume keys: the volume up key to move to the previous character and the volume down key to move the next character.

The granularity for reviewing a text can be changed through the local context menu which contains controls that relate to the focused item, so here, the local context menu for a text field. To launch the menu, the user swipes up and then

[1] `https://support.google.com/accessibility/android/answer/6283677?hl=en&ref_topic=3529932`

[2] `http://www.apple.com/accessibility/ios/voiceover/`

right. A circular menu appears and the user has to drag his finger in a circle over the menu to hear the menu items. For a text field, it will basically be granularity and selection settings. Indeed, for instance the user can change the default granularity from character by character to word by word or select the whole text to erase it. A global context menu is also available to change global settings as text-to-speech settings or the luminosity of the screen, it works the same as the local context menu, except that the manipulation must be carried out when non-specific components are focused. These menus can be complex to use as confirmed during the user studies presented in section 5.2.

2.3.4.2 VoiceOver

VoiceOver proposes a simple set of gestures to let the disabled user easily control his iOS device. Among these gestures, if the user touches or drags his finger around the screen, VoiceOver tells him directly what is there. To hear a description of a button, the user has to tap; and then, double-taps to activate it. Swipes up or down are used to adjust a slider. To move from one app to the next, the user just flicks left and right. When the user interacts with an element on the screen, a black rectangle appears around it, so sighted users can follow along. An interesting setting that enhances privacy is the possibility to activate a screen curtain to turn off the display so no one can see it, even as the user is controlling it.

Regarding text input, VoiceOver echoes each character on the keyboard as the user touches it, and again when he enters it. To edit precisely, the user has to flick up or down in order to move the cursor.

Moreover, iOS devices are fully compatible with many refreshable Braille displays and Braille displays with input keys can be used to control your iOS device when VoiceOver is turned on. VoiceOver also includes a systemwide Braille keyboard that supports Braille chords in 6 and 8 dot Braille, enabling direct Braille entry without the need for a third party physical Braille keyboard.

Chapter 3

Fission & Multimodal Output

The aim of this chapter is to explore the output part of the multimodal interaction and more precisely the *fission*. The use of the terms will also be covered to know if it is still used or not and if not, trying to identify by what it was replaced. First, a global definition is given and will be used as a reference to make comparisons with what it is presented in the papers gathered in the second section of this chapter.

Contents

3.1 Main Definition

Foster [20] defined the fission as followed: *"In multimodal interactive systems, fission is the process of realising an abstract message through output on some combination of the available channels."*

Moreover, the fission can be divided in 3 phases:

1. The ***content selecting*** and ***structuring***: the information to deliver at the user must be selected and structured regarding an overall structure in a way that it can be conveyed through the different modalities;

2. The ***modality selection***: this phase determines which modality or combination of modalities is the best to represent the output regarding the context;

3. The ***output coordination***: a coherent presentation requires a coordination from the different modalities. For instance, for an intelligent agent, the speech should be coordinated with the head's movement.

3.2 State of the Art on Multimodal Fission

This section presents a few scientific papers related directly to the fission or more generally to the output part of multimodal systems. These articles are ordered chronologically.

3.2.1 A Standard Reference Model for Intelligent Multimedia Presentation Systems [7]

In 1997, Bordegoni *et al.* presented the SRM (**Standard Reference Model**) which is a model for intelligent multimedia presentation systems based on five layers (from the top to the bottom):

- ***Control*** *Layer* to allow the selection of the goal that the system must achieve;

- ***Content*** *Layer* to refine the goal into several specialized sub-goals and then for each of them, to select the adequate modality, media and presentation content;

- ***Design*** *Layer* to set the morphological (as font size) and spatial-temporal (layout and timing) attributes of the presentation;

- ***Realization*** *Layer* to generate the effective presentation;

- ***Presentation*** *Display* to distribute the different components of the presentation to the appropriate media and to coordinate the various components in order to construct the global presentation.

Basically, the output interaction is built from a precise goal that consists of the information to be presented to the user. Then the goal is refined into sub-goals and their morphological and spatial-temporal attributes are determined. Finally, the multimodal interface output presentation is generated and is distributed to available media.

3.2.2 WWHT: A Conceptual Model for the Multimodal Information Presentation [38]

This paper of 2005 presents a life cycle conceptual model based on the SRM called **WWHT** which stands for ***"What"***, ***"Which"***, ***"How"*** and ***"Then"***. To each proposition corresponds to a question:

- ***What*** information has to be presented?
- ***Which*** modalities to choose for the presentation?
- ***How*** to present this information with the chosen modalities?
- ***Then*** how to handle the evolution of this presentation?

Answering these questions results in the following architectural design process where each step answers one of the four questions:

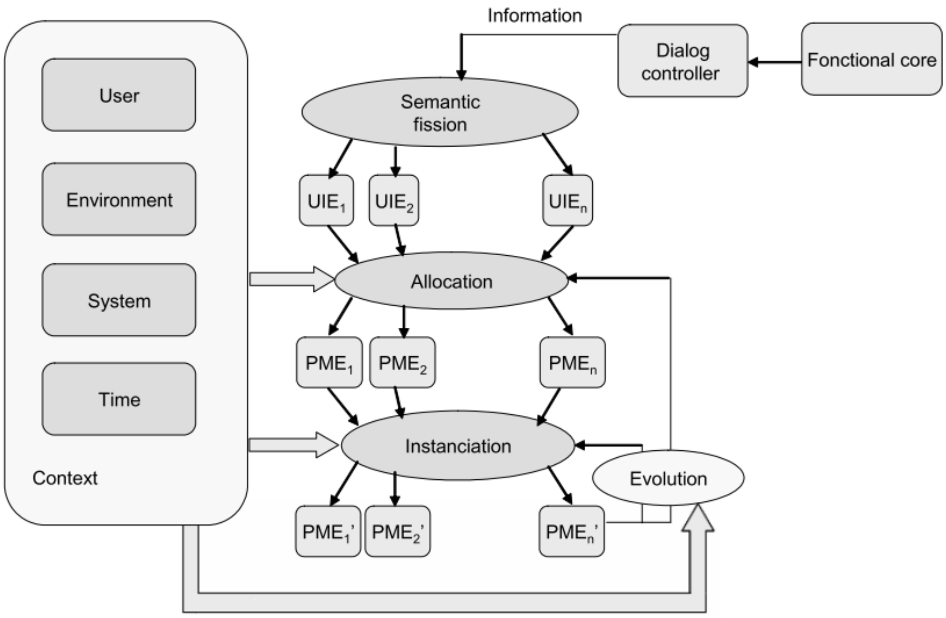

Figure 3.1: The WWHT model [30]

Indeed, the ***semantic fission*** decomposes the semantic information issued from the dialog controller into elementary units of information, which is the answer to *"What?"* and correspond to the first phase of the fission presented by Foster. The authors also said that for them it is not pertinent to use the term *"fission"* as opposite of the term *"fusion"* to define the whole process of building the output presentation. But as a fission actually happens at the semantic level, they prefer to use *semantic fission* only for the decomposition of the information.

The ***allocation*** consists in allocating to each unit of information the right modality regarding the interaction context and consolidate it into a global presentation consistent with the initial information provided. It is the answer to

"Which? " and it is basically the second phase presented by Foster.

The ***instantiation*** determines the concrete contents to be displayed through the chosen modalities and sets their attributes as well as spatial-temporal attributes, it answers the *"How?"*.

The ***evolution*** step answers the "*Then?*" which is not present in the classical Foster's definition of the fission and takes care of the possible evolution of the presentation. This concern reflects the assessment that the context of use is constantly evolving and so a presentation built in a specific context may not be adapted after some changes in this context.

3.2.3 Multimodal Output - Specification / Simulation Platform [37]

Rousseau *et al.*[37], in their paper of 2005, mentioned that the emergence of new computer systems for the general public, like mobile or PDA needed a new approach to specify complete and valid output interaction. More precisely, it is a transition between a situation where the interaction had to be adapted to a given application and in a particular context to a situation where the context constantly evolved with multiple platforms applications.

As suggested by the title of the paper, they came up with a platform that allows respectively the output specification and the output simulation of a multimodal system. For that, they defined their own data representation language based on XML to describe the content and they used election rules to describe the behavioural model of the interface.

3.2.4 The SmartKom Architecture: A Framework for Multimodal Dialogue Systems [23]

In 2006, Herzorg and Reithinger presented SmartKom which provides an adaptive and reusable dialogue shell for multimodal interaction. As shown on the figure 3.2, they proposed the following conceptual architecture based on the functional blocks of a generic multimodal dialogue system and decomposed the processing task into the presented components.

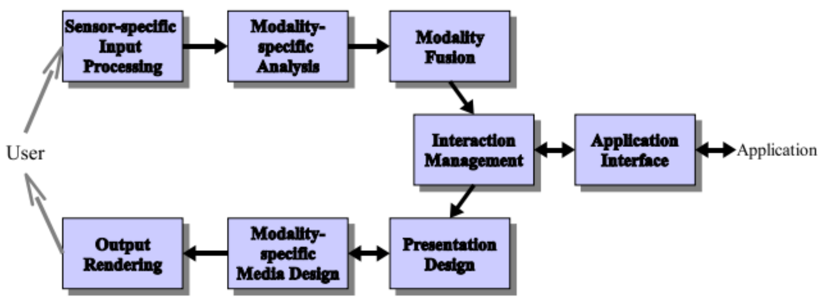

Figure 3.2: A high-level architecture for multimodal interaction [23]

The components that are involved in the output side are defined as following by the authors:

- ***Interaction Management***: Here resides the mind of the machine. This part of the system identifies the intention of the user, determines the next steps to be taken by the system and addresses the various application functions;

- ***Presentation Design***: Given a system intention, the information to be conveyed needs to be transformed into coordinated multimodal output. This functional block constitutes the first stage of modality fission and deals with the overall organisation of the intended multimodal presentation. It includes in particular sub-tasks like content selection, media and modality allocation, layout design and coordination;

- ***Modality-Specific Media Design***: In close cooperation with the presentation design stage, the processing of this layer is specifically concerned with the conversion of abstract content structures into presentable media objects;

- ***Output Rendering***: The final function block comprises the technical means to present the coordinated system reaction in the defined media channels like screens, loudspeakers or force-feedback devices.

The main advantage of SmartKom is that it is based on an open architecture for multimodal dialogue systems that is flexible and adaptive. Indeed, all available components can be reused directly and modular organisation into different service components which simplifies the integration of application-specific functions.

3.2.5 Physical, Semantic and Pragmatic Levels for Multimodal Fusion and Fission [28]

As mentioned in the paper's title, Landragin [28] presented in 2007 three levels where the information can be split during the multimodal fission:

- At the ***physical*** or ***signal*** level, the information, considering its nature, is sent directly to the correct communication channel. This is strongly linked to the constraint-based repartition over the communication channels. This process is called ***multimedia coordination***;

- At the ***semantic*** level, the information content can be dissociated over several modalities in order to better manage its complexity and to simplify the resulting monomodal messages. This is linked to a preference-based repartition. The author talked of ***content fission*** for this process;

- At the ***pragmatic*** level, the concept of ***presentation act fission*** is introduced. The message illocutionary force (locution is what was said, illocution is what was meant) can be dissociated over several modalities in order to simplify the illocutionary force of each monomodal message. The following example is given: an informative message that needs an acknowledgement of receipt from the user can be split into two messages; a first one that verbalizes the *"inform"* and a second one that *"demands"* the acknowledgement using a text box.

3.2.6 A Generic Formal Model for Fission of Modalities in Output Multimodal Interactive Systems [30]

Oussaïd *et al.*[30] in their paper of 2009 tried to address the following problem: the sequential and parallel combination of modalities increases the complexity of the information representation. And so, this requires more complex development as well as validation processes. The paper focused on formal specifications of the multimodal output interaction.

The fission model presented expresses the semantic fission of the WWHT model with a description of a specific syntax as well as the static and the dynamic semantics thanks to temporal and semantic operators. The CASE space presented in section 1.2.1 is used to parametrize this model according the four different classes of system presented in it. So the proposed approach offers a generic design model and a parametrized one according to the CASE space.

3.2.7 Adapting Multimodal Fission to Users Abilities [14]

In 2011, Costa and Duarte [14] motivated a system which uses multimodal fission to infer an adequate user interface for elderly and differently impaired users. The presented work was developed in the scope of the European *GUIDE* (*G*entle *U*ser *I*nterfaces for *E*lderly *P*eople) project. Their fission process is also based Rousseaus WWHT tasks. They planned to realize a rule-based composition approach.

They also came up with the fact that *"there is not much research done on fission of output modalities"*. They gave the following possible explanation to this shortage: this is because most applications use only few different output modalities, and therefore use simple and direct output mechanisms.

3.2.8 Adaptive Probabilistic Fission for Multimodal Systems [24]

In this paper of 2012, Honold *et al.* [24] presented *ProFi*, a system for *Pro*babilistic *Fi*ssion, designed to reason on adaptive and multimodal output based on uncertain or ambiguous data.

Indeed, system are usually not very smart in how they offer and communicate their functions. On the contrary, human beings own the ability to reason about the way they express themselves, they inspect their surroundings, judge the information to be communicated, and monitor their communicative counterpart.

The idea proposed by the authors is to allow multimodal system to reason in the same way that human did by using a probabilistic based reasoning approach for modality arbitration. Indeed, this approached allowed to integrate uncertain and ambiguous knowledge.

3.2.9 Multimodal Fission for Interaction Architecture [46]

In 2013, Zaguia *et al.* proposed a modular and distributed architecture which contains five main modules as shown in the figure 3.3 here below.

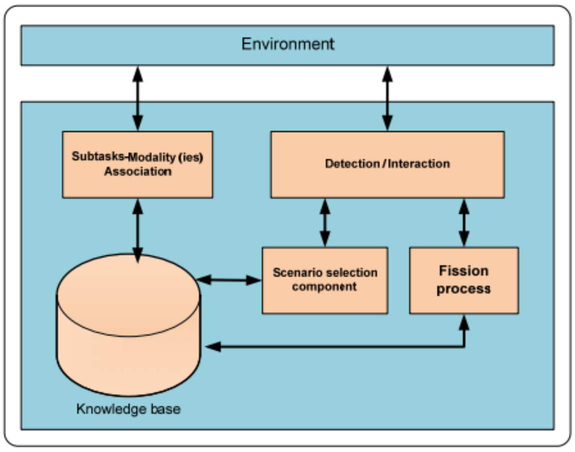

Figure 3.3: Multimodal fission system architecture by Zaguia *et al.* [46]

The 5 components as defined by the authors as followed:

- ***Detection/Interaction***: This module will interact with the environment to allow modules "Fission"process and "Scenario selection component"to achieve fission. It detects any variation in the environment, for instance the change of the noise level that affects the selection of the audio modality;
- ***Fission process***: It represents the fission rules/algorithm required to realise the fission.
- ***Subtasks-Modality (ies) Association***: By using patterns, this module allows to select scenarios that occurred previously and stored in the knowledge base;
- ***Modality selection***: This module allows selection of the appropriate modalities available for each sub-task;
- ***Knowledge base***: It describes the environment, the modality patterns and the patterns of scenarios that occurred previously.

This architecture is quite comparable with the other ones presented in this section but differs by the fact that the fission module focuses on the use of patterns. Patterns are predefined models that describe a selection of one or several modalities. For them, a modality pattern is composed of two elements: a *problem* himself decomposed in 6 subcomponents (application, parameter, priority, combination, scenario and service) and a *solution* composed of the chosen modalities.

3.2.10 A Formal Model for Output Multimodal HCI - An Event-B Formalization [31]

Oussaïd *et al.*[31] in their paper of 2015 motivated their approach by the need of rigorous and formal methodology to develop multimodal interfaces in critical contexts like the health sector or the nuclear sector. Indeed, critical systems need particular attention to ensure safety and the conformity with the design requirements. This paper is in the direct continuity of *A generic formal model for fission of modalities in output multi-modal interactive systems*.

Again, the fission is also mentioned as being only the first step of the multimodal output process. Two formal models are presented :

1. *"The fission model describes the semantic decomposition of information produced by the functional core into elementary information"*

2. *"The allocation model specifies the allocation of modality/media pairs for each elementary information"*

Theses models have been formalized with Event-B allowing developers to achieve properties validation. Moreover, this formal approach is proof-based and supports the design of large systems. Theses models are based on the WWHT model presented earlier which was himself based on the standard reference model (SRM) by adding the evolution phase *"Then"* as said before. But unlike the models presented in this paper, SRM and WWHT are semi-formal models and so are not sufficient to ensure complete correctness.

3.2.11 Conclusion

Through these ten scientific papers from 1997 to 2014, it appears that some concepts have been reused over the years. Indeed, the SRM has inspired the WWHT model which himself was reused in some of the other presented papers.

Regarding the use of the term fission, it appears that, as mentioned by Rousseau *et al.* [37], it is better to use it to represent the decomposition of information where a *"fission"* actually happens and not to define the whole process of building the output presentation. This preference was found in some of the various presented papers. Moreover, the term *"multimodal output presentation"* himself seems to be more popular to describe the whole process.

In 2011, Costa and Duarte [14] came up with the following fact: *"there is not much research done on fission of output modalities"*. Their explanation for this shortage was that most applications use only few different output modalities, and therefore use simple and direct output mechanisms. This assessment seems to be still the case in 2016. Indeed, finding papers where the output part of multimodal system is developed can be challenging and most of them focuses on the input side. This is also confirmed by simply looking at the number of citations of the report of Foster [20] which is only 30. Despite its release date (2002), this report is still currently the most complete document on fission and has even never been published. This proves well that the output part of multimodal systems needs more attention from the scientific community.

Chapter 4

A Conceptual Architecture to Enhance the Smartphone's Accessibility for Visually Impaired People

Mobile devices accessibility for users with visual impairments has improved remarkably in recent years. They are now able to perform many tasks independently on their smartphones, supported by accessible interfaces provided by the main OS developers. Googles TalkBack for Android and Apples VoiceOver for iOS (presented in section 2.3.4) are solutions that have empowered users with visual impairments to use and enjoy their mobile devices at a level that could not be imagined recently. But these solutions are mainly screen exploration techniques for navigation. Moreover, they are mostly beneficial in controlled environments, like their home or office. Uncontrolled environments (like the street) still raise a number of challenges that prevent this user group from operating their devices at a similar level. Blind users typically have to stop operating their devices and they usually refrain from doing so in the middle of the street. This chapter aims to present a model to use the smartphone functionalities in a more adequate way regarding the disabilities of visually impaired people.

Contents

4.1 Main Idea

The main idea is to directly access the desire functionality through short-cuts and so, to reduce the time that a user spends navigating the user interface to reach this functionality. Indeed, user interface navigation represents a significant percentage of the interaction time for the visually impaired.

These short-cuts will be assigned, for example, to a chosen gesture which allows the user to directly interact with the phone without touching it. Multi-modality is central in this concept as it allows to interact in a more human way as presented in section 1.2.

4.1.1 Conceptual Architecture

A simplified architecture should have the following components:

- ***Input and Short-Cuts Manager***: This component has to manage the recognizing of a short-cut when it is executed and then send an event to the functionality's directory to launch the corresponding functionality. It also has to take care of the input required for the different functionalities, the different modalities presented in section 2.2 can be used regarding the context captured by the context manager. It could also communicate with the output and feedback manager to send feedback regarding the validation of the input to the user.

- ***Functionality's Directory***: It regroups the different functionalities available and reacts when it receives an event from the short-cuts manager. If the functionality chosen requires more input, it will dialogue again with the input manager to get all the mandatory data to process and execute the asked command. Then, it will communicate with the output and feedback manager to inform the user of the results.

- ***Output and Feedback Manager***: This component is meant to show the result of the functionality's call if there is one concrete result to show or simply furnishes basic feedback. It will also communicate with the context manager in order to choose a modality that fits the most to the user's context.

- ***Functionality's Extractor***: This component has to capture and extract functionalities in order to put them in the directory. For instance, as presented just after in section 4.2, it could simply extract the information about existing user interface and then directly interact with them. But it is not unreasonable to think that this vision can be broadened and generalized to capture functionalities directly from an interface or service and then generate some code. Indeed, with the emergence of the *API Economy*, more and more application programming interfaces (APIs) are available. According to Collins and Sisk [12] in 2015, the public APIs had doubled in the 18 past months and more than 10,000 APIs have been published to this date. Therefore, this means an easy direct access to functionalities without the need to pass through a user interface.

- ***Context Manager***: This component is in charge of capturing a maximum of information about the context (as defined in section 1.3) of the user through the different sensors and technologies embedded in the smartphone. For instance, his localisation, the noise level, if the user is moving or not, etcetera.

The different interactions between the components are summarized in the figure 4.1 here below.

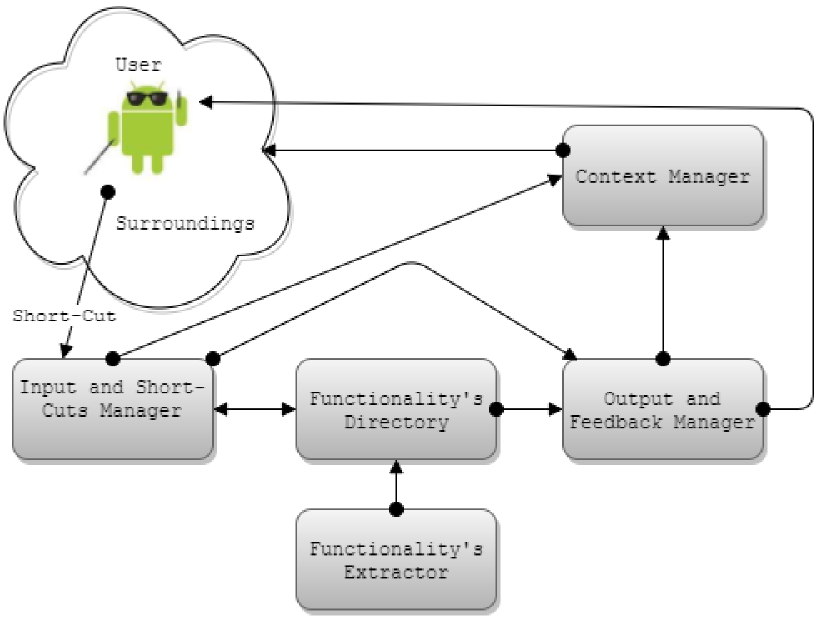

Figure 4.1: Proposed Conceptual Architecture

4.2 Use Case: An Android Accessibility Service to Capture Interface Components

The *Accessibility API* provides features to build services that make other applications more accessible for users, these services are called ***Accessibility Service***. An accessibility service is defined by Android as : *"An accessibility service is an application that provides user interface enhancements to assist users with disabilities, or who may temporarily be unable to fully interact with a device. For example, users who are driving, taking care of a young child or attending a very loud party might need additional or alternative interface feedback."* [1]

One accessibility service was developed to retrieve and interact with the component of the interface and to save this structure in a standard format to be able to transmit it to other systems. Regarding the model defined before, it is the same as a functionality's extractor and directory.

[1] `http://developer.android.com/guide/topics/ui/accessibility/services.html`

4.2.1 Global Working

An accessibility service will react at the reception of ***AccessibilityEvents*** These events are sent by the operating system when something remarkable occurs on the screen: clicking, long clicking, selecting an item in a list, focusing on a component, changing the text of a field, changing the text selection of a field, traversing the text of a view at a given granularity, scrolling in the application or in a list, opening a new window or a menu, changing the content of a window, when a notification is shown, starting a touch interaction, ending a touch interaction, etcetera. Of course, it is possible to filter which type of events the accessibility service is supposed to react to, but also to choose within which applications the service must run.

The most important methods of an accessibility service is the *onAccessibilityEvent()*, it is the method that is called back when an event that matches the filtering parameters chosen is detected. But more importantly, this method allows to extract the component from the event that is responsible for the event's emission (through the *AccessibilityEvent.getSource() method*) which was not the case in the early version or to get more information about the context of the event's origin (through the *AccessibilityEvent.getRecordCount()*) method.

The component retrieved from the event will be in the form of an *AccessibilityNodeInfo* object. As indicated by its name, this object is a node which is part of a tree structure. Therefore, it is possible to go back up to the root of this tree and then from the root, to navigate in the opposite way through all the possible paths to build the tree of all the elements of the screen.

Once this tree is built, it is possible to perform certain actions on a node thanks to the *performAction(int action)* and *performGlobalAction(int action)* methods. So as the denomination of the methods indicates, this action can be *local* (clicking on a button, scrolling a list, ...) or *global* (navigating to the home screen, push the back button and opening the notification screen or the recent application list).

It was also decided to save the tree structure into a standard format, the UIML[2] format was chosen. The UIML file is written at the same time that the tree is built by the accessibility service. The generator was developed with the Document Object Model (DOM) API of Java. Examples of UIML files can be found in section 4.2.2. For each component, its class and its content description is written in the file as well as an ID which is generated during the creation of the file. The purpose of the ID is the let other components, for instance an in-air gesture recognizer, communicate with the accessibility service through an event filled with the ID of the component and the action to be performed. Indeed, it is possible to use the accessibility event of type *AccessibilityEvent.TYPE_ANNOUNCEMENT* to send customized accessibility events.

[2] `http://docs.oasis-open.org/uiml/v4.0/cd01/uiml-4.0-cd01.html`

4.2.2 Example of Use

This section presents the capture of two different interfaces by the developed accessibility service. One is the interface of the home screen of a locked phone and the other one is the interface of a small home-made application. They are shown on the figure 4.2 here-bellow :

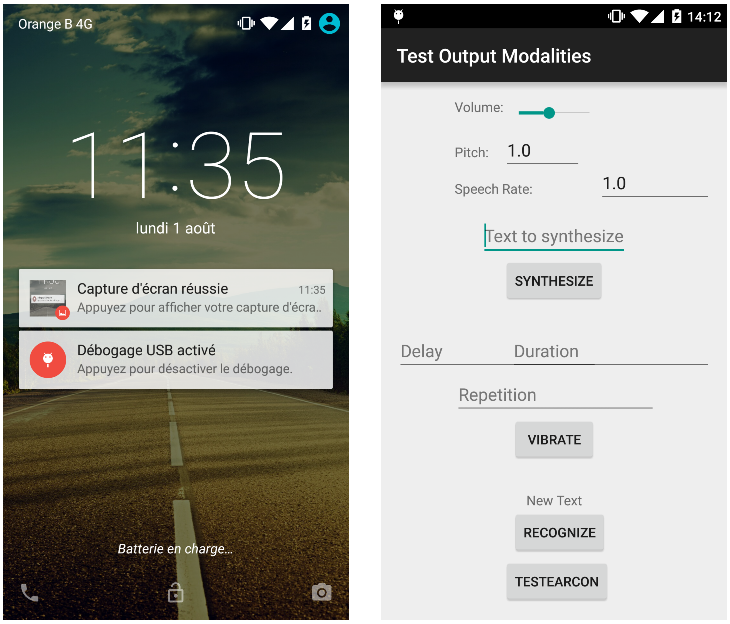

Figure 4.2: Interface of a locked home screen on the left and interface of a small home-made application on the right

Here is the UIML code of the home page interface, all the components are well present and it can be noticed that the notifications are grouped in another component that has no text or content description since it is null. Indeed, the field *ContentDescription* is filled in with the actual content description field of the component if it has one, if not with the text field of the component and with null otherwise.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE uiml PUBLIC "-//OASIS//DTD_UIML_4.0_Draft//EN" "http://uiml.
    org/dtds/UIML4_0a.dtd">
<uiml xmnls="http://docs.oasis-open.org/uiml/ns/uiml4.0">
  <interface>
    <structure>
      <part class="android.widget.FrameLayout" id="tonzaacj91t0" ContentDescription="
          Batterie_en_charge">
        <part class="android.widget.TextView" id="1iykj45si6ixu" ContentDescription="
            Batterie_en_charge"/>
        <part class="android.widget.ImageView" id="1jewm0ixd6yqq" ContentDescription
            ="Tlphoner"/>
        <part class="android.widget.ImageView" id="1jyobhqfb5m47" ContentDescription
            ="Dverrouiller"/>
        <part class="android.widget.ImageView" id="1jeqkjgmx3wwy"
            ContentDescription="Appareil_photo"/>
        <part class="android.view.View" id="1ib60vt1a3syt" ContentDescription="null">
```

```
        <part class="android.widget.FrameLayout" id="1iyc7s8u8zqea" ContentDescription="Capture d'cran russie"/>
        <part class="android.widget.FrameLayout" id="1hux7ogvbogh2" ContentDescription="Dbogage USB activ"/>
      </part>
      <part class="android.widget.TextView" id="1iukvl5fcye5z" ContentDescription="Orange B 4G"/>
      <part class="android.widget.ImageView" id="somwhchqw9pz" ContentDescription="Sonnerie en mode vibreur"/>
      <part class="android.widget.FrameLayout" id="t4yzbkb94w8k" ContentDescription="Signal Wi-Fi excellent"/>
      <part class="android.widget.FrameLayout" id="qhkkcc1tkgv7" ContentDescription="null Signal excellent"/>
      <part class="android.view.View" id="qdycht2qc7s5" ContentDescription="Battery 98 percent."/>
      <part class="android.widget.TextView" id="uv8n2zoojq02" ContentDescription="lundi 1 aot"/>
      <part class="android.widget.TextView" id="1hrm0l4imomk5" ContentDescription="11h35"/>
    </part>
  </structure>
 </interface>
</uiml>
```

In the code of the interface of the home-made application, the actual ContentDescription field of some components were modified regarding the state of the component. For instance, on lines 11 and 13, the values are *"Pitch Value 1.0 default value"* and *"Speech Rate Value 1.0 default value"* and not simply the value of the textfield if nothing had been put in the ContentDescription field.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE uiml PUBLIC "-//OASIS//DTD UIML 4.0 Draft//EN" "http://uiml.org/dtds/UIML4_0a.dtd">
3 <uiml xmnls="http://docs.oasis-open.org/uiml/ns/uiml4.0">
4  <interface>
5   <structure>
6    <part class="android.widget.FrameLayout" id="1hrlzf5hzraea" ContentDescription="Test Output Modalities">
7     <part class="android.widget.TextView" id="rhakw80bnl0x" ContentDescription="Test Output Modalities"/>
8     <part class="android.widget.TextView" id="skydmgvv0qjn" ContentDescription="Volume:"/>
9     <part class="android.widget.SeekBar" id="sl668jir6b74" ContentDescription="VolumeSeekBar 0-100"/>
10    <part class="android.widget.TextView" id="1hrbfcrpybdwz" ContentDescription="Pitch:"/>
11    <part class="android.widget.EditText" id="1ji7auty7nprp" ContentDescription="Pitch Value 1.0 default value"/>
12    <part class="android.widget.TextView" id="1iesqnrdp2eed" ContentDescription="Speech Rate:"/>
13    <part class="android.widget.EditText" id="t89mj07aouhv" ContentDescription="Speech Rate Value 1.0 default value"/>
14    <part class="android.widget.EditText" id="rhr5zghrc20i" ContentDescription="Text to synthesize"/>
15    <part class="android.widget.Button" id="sl3gc6jc9yzn" ContentDescription="Synthesize"/>
16    <part class="android.widget.EditText" id="us2zjiy387qp" ContentDescription="Delay"/>
17    <part class="android.widget.EditText" id="t87d2gxpf885" ContentDescription="Duration"/>
18    <part class="android.widget.EditText" id="1h84tk9qja4py" ContentDescription="Repetition"/>
```

```
        <part class=”android.widget.Button” id=”s1b5krefa055” ContentDescription=”Vibrate”/>
        <part class=”android.widget.TextView” id=”r1ifkuttq0xh” ContentDescription=”New Text”/>
        <part class=”android.widget.Button” id=”qhkl44f1mbld” ContentDescription=”Recognize”/>
        <part class=”android.widget.Button” id=”s1efzlcuj42u” ContentDescription=”TestEarcon”/>
      </part>
    </structure>
  </interface>
</uiml>
```

4.2.3 Limits & Problems Encountered

Unfortunately, the accessibility services are quite limited. This section presents some issues and limitations encountered when using the Accessibility API.

Firstly, the manipulation of components in the form of *AccessibilityNodeInfo* objects limits the possible processing on them. Indeed, the captured component has to be manipulated through the methods of this object's type and not the ones of its specific type. For instance, in the case of an object of type *ListView* used to display a list, when it is captured by the accessibility service, it is not possible to have access to its *Adapter* and so there is no means to interact or retrieve all the elements of the list. Some issues also occurred with the objects of type *TextField* where its content cannot be modified inside the service.

Secondly, the only information available on a component is the one provided by the developer in the *ContentDescription* field of the component. If it is empty or non-explicit, it makes the identification of the component's purpose impossible. This description varies as well in function of the phone's constructor or the language of the phone. The difference can be faint, for instance, the description of the *ImageButton* to end a call was *"EndCall"* when ran on the emulator and *"End call"* on another physical smartphone. How small the difference is, a special attention has to be taken to make the correct processing to ensure that the system is operational on different smartphones.

Thirdly, as its name suggests, an accessibility service is a *Service*[3] and so it is meant to be executed in the background with no interaction with the user on the contrary of an *Activity*. Indeed, a service is defined as : *"A Service is an application component that can perform long-running operations in the background and does not provide a user interface. Another application component can start a service and it will continue to run in the background even if the user switches to another application. Additionally, a component can bind to a service to interact with it and even perform interprocess communication (IPC). For example, a service might handle network transactions, play music, perform file I/O, or interact with a content provider, all from the background."*

Thus, it is not possible to call the *startActivityWithResults()* method from an accessibility which is mandatory to launch certain functionalities such as the speech recognizer. A possible not advised solution is to make the call from an

[3] https://developer.android.com/guide/components/services.html

intermediary activity which has to have a layout. This layout can be empty but in certain case, such as the use of the speech recognizer, a button is required to be able to make the call. It is not recommended because normally a window that pops from nowhere will probably annoy the user more than anything even if here, it may be less annoying for the targeted users. Other manipulations such as saving a file on the external part of the memory within a service is also more difficult than within an activity.

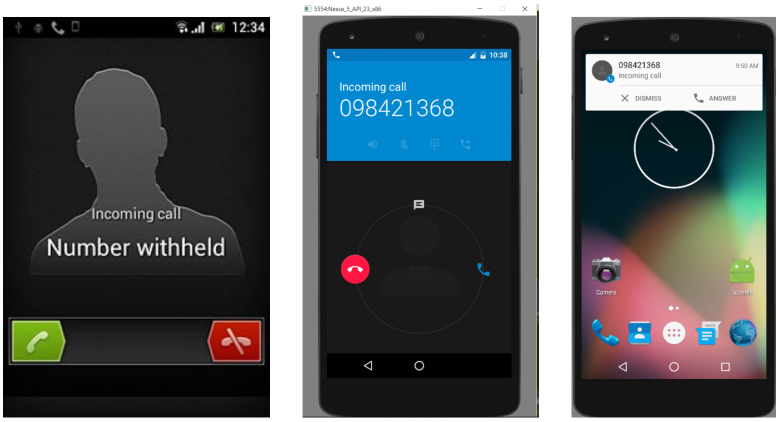

Figure 4.3: Possible interface to answer a call

Finally, not only the *ContentDescription* field varies according to the phone's constructor, there can be some interfaces specific to the constructor layer or to the Android's version of the phone. For instance, when receiving a call, it can be a slider button (see the left picture of the figure 4.3) that hides 2 choices (accept or decline) or a button to extend (see the middle picture of the figure 4.3) that offers 3 choices (accept, decline or answer with an SMS). This interface can also vary whether or not the phone is locked when the call occurred, as shown on the right picture of the figure 4.3. This also raises the difficulty to create a non-specific service to a certain phone's model in particular.

Chapter 5

Case Study: An SMS Application

Using a text message service is one of the most basic and most used application when using a mobile phone. This chapter presents a prototype based on the model presented before, this prototype is an SMS application using in-air gestures and voice recognition for the Android platform to directly perform SMS related tasks. Unfortunately, current solutions of gesture recognizers, like those presented in section 2.2.4.1, are still not capable to reliably address a per-user on-the-fly customizable set of gestures. Therefore, for the prototype it was decided to use the Wizard of Oz approach [22] in the user studies. Gestures made by study's participants are interpreted by a human that inputs the command in the application. Voice commands are interpreted by the default speech recognizer available in the Android API. Nevertheless, it would have been possible to do it with predefined gesture and the accessibility service developed presented in section 4.2.

Contents

5.1 General Presentation

The application developed offers the four following standard SMS service features:

1. The user can enter a message;
2. The user has the possibility to review and edit the message if necessary;
3. The user can find and select the message addressee;
4. The user can send the entered message to the selected addressee.

5.1.1 Implemented Functionalities

To perform these features, different actions are available to the user. Each action is triggered by a specific gesture and so it can allow the user to manipulate the application without having to hold the smartphone in their hands. The possible actions proposed at the user to guarantee the features explained before are the following:

- **Add a message**: After recognizing the *"add a message"* gesture, the speech recognizer is turned on and the user is notified by a sound. Then, the user dictates the message.

- **Review the message**: After recognizing the "review message"gesture, the text-to-speech engine is turned on and the message is read word by word.

- **Edit a specific word in the message**: During the message review, the user can stop the speech engine (by again performing a specific gesture) whenever he wants to edit a word. Whenever that gesture is recognized, the application speaks the chosen word and turns on the speech recognizer. The user then dictates the new word. After that, the system informs the user that he will have to choose between five proposals. The application reads the proposals and the user stops it (by making a specific gesture) whenever he wants to select the desired proposal. The application tells the selected proposal and the change is carried out in the message.

- **Enter a token to search for a contact**: After recognizing the *"enter contact"* gesture, the speech recognizer is turned on and the user says the name of the contact or a word that begins with the same letter of the contact's name. Then the application speaks how many contacts match this search. The user can add other tokens to refine the search or proceed to hear the names returned by the search.

- **List the contacts that match the current search**: The application reads all the contacts that match the search token previously entered and the user can stop the reading to select the desired contact. The application orally notifies the user of the chosen contact.

- **Send message**: After recognizing the *"send message"* gesture the message is sent and the user is notified.

All these actions are alternate tasks as presented in the CASE model in section 1.2.1. Indeed, to complete each task, it requires a temporal alternation of modalities (speech and gesture) using coreferences.

Two languages (English and Portuguese) were also proposed for the application. Switching from one to another changes the language's setting of the speech recognizer and of the text-to-speech, the names of the contacts as well as the feedback messages are also translated.

5.1.1.1 Interface & Technical Details

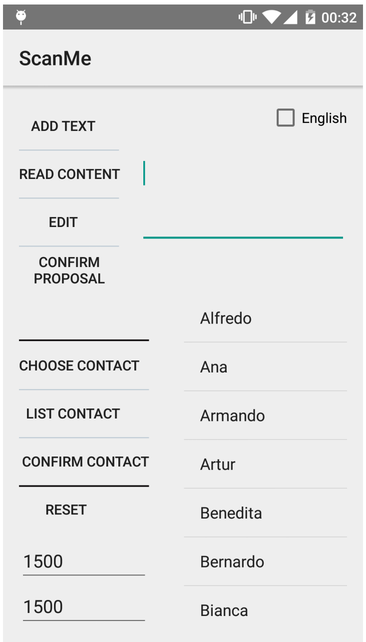

Figure 5.1: Interface of the developed prototype

As shown on the figure 5.1, a set of buttons on the left side trigger the different possible commands. The two textfields on the left bottom corner are used to change the delay between two words respectively for the reviewing and for the listing of contacts. The checkbox on the top right corner allows to change the language (English when it is checked, Portuguese otherwise). The right side of the interface is just used as support to verify the validity of the message and to have an overview of the contacts' list.

The developed application relied on the text-to-speech provided by the Android API[1] and the vocal recognizer was launched through a *RecognizerIntent*[2].

[1] http://developer.android.com/reference/android/speech/tts/TextToSpeech.html
[2] http://developer.android.com/reference/android/speech/RecognizerIntent.html

Unfortunately, when used directly in the mean thread, the text-to-speech froze the application and so, it was no longer possible to click on any button. Therefore, a specific thread was created to ensure that the application is still functional when the text-to-speech is on. This thread also allowed to interrupt and resume easily the reading but, first of all, to retrieve the word that is currently read which is really useful during the edition and the contact picking.

Regarding the search of a contact, a simple prefix based search algorithm was used to isolate every contact where the entered token(s) is(are) the prefix of this contact in order to be able to tell the user how many contacts match his search.

5.1.1.2 Rationale Behind Some Design Decisions

The design decision of presenting five proposals during the edition is to counter possible miss-recognition of the speech recognizer. Actually, it is possible that the user wants to edit a word that was miss-recognized and the right word has a high probability to be in the first five possible proposed recognition. Thus, it avoids that the first wrong proposition would be chosen every time just because the pronunciation of the user is not perfect.

The same motivation was behind the decision to consider only the first letter when entering a search token. Moreover, here it is highly probable that if the user tells a specific first name or last name that it would be miss-recognized. But the first letter will probably be correct and if not, the user can just say another word that would be normally recognized correctly as a country name for example; this can even enhance the privacy.

A special attention was given to the message editing process to make it more accessible. Indeed, as mention in section 2.2.1.1, a survey of Azenkot *et al.*[4] showed that people with visual impairment spend 80% of their time editing when using speech recognition which can be frustrating.

5.2 User Studies

To evaluate the prototype, two users studies were performed. The first one was done with sighted users and the second one with unsighted users. The complete user study preparation can be read in the appendix A. Theses studies have two goals. The first one was to compare, performance-wise, the use of the prototype with a standard SMS application coupled with VoiceOver or Google TalkBack. The second one was to observe and understand the behaviour of visually impaired people as the type of gestures they like to perform and if they would feel comfortable using such system on a daily basis.

5.2.1 Global Proceedings

First, a few questions were asked to the participant, this questionnaire includes personal information as well as mobile phone usage habits. Thus, the following information about the participant was gathered:

- His age;
- His gender;
- For how long does he own a smartphone?
- How expert does he classify himself in using TalkBack or VoiceOver?
- How often does he send SMS via his smartphone?
- How expert does he classify himself in using the SMS application or a similar message service?

Some additional questions were asked at the unsighted participant:

- When did the participant lost his vision?
- What type of software does he use for sending SMS?
- What does he do when he receives a call or SMS on the street, does he stop answering the call or does he wait until he reaches his destination?
- Will he use in-air gesture to interact with his smartphone?

After gathering this information, it was randomly chosen if the participant will start the evaluation with the developed application or with the standard application. Participants were then asked to perform the same set of tasks in both the prototype and the usual messaging application. To compare their performance, a quantitative evaluation was designed based on time to perform different tasks on both applications. Three different levels with increasing difficulty were defined to create the tasks:

- ***Easy*** task: The participant is asked to enter a text message (the message was provided). The participant is requested to review the message entered. No edition of the message is requested. The addressee is easily findable (maximum 3 results in the prototype condition and a contact figuring in the beginning or the end of the contact list for the standard application);
- ***Normal*** task: The participant is asked to enter a text message (the message was provided). The participant is requested to review the message entered. The edition of 1 word of the message is asked. The addressee is moderately easy to find (maximum 5 results in the prototype condition and a contact figuring in the first or last quarter of the contact list for the standard application);
- ***Hard*** task: The participant is asked to enter a text message (the message was provided). The participant is requested to review the message entered. The edition of 2 words of the message is asked. The addressee is more difficult to find (more than 5 results in the prototype condition and a contact figuring in the middle of the contact list with a name beginning with a frequent letter for the standard application).

Each task was timed by an observer, who registered the time each task and sub task (writing message, reviewing the message and editing it if necessary) took to be completed and also the errors made.

An evaluation concerning the delay between the reading of two words during message review or two contacts was also performed. Participants were asked to choose between 3 possible delays (**1**s, **1.5**s and **2**s) the one preferred. By default, during the other tasks, the value was set at 1.5s. The order of presentation of the delays was randomized. The purpose was to understand how fast they are able to react and perform the gesture to select a word or contact.

The final step consisted in filling 2 **SUS** (**S**ystem **U**sability **S**cale) based satisfaction questionnaires: one for the standard application (only done for the unsighted users) and one for the prototype. This type of questionnaire was originally proposed by Brooke [10] and the SUS consists of the following statements that the user rates from 1 (strongly disagree) to 5 (strongly agree):

- "I think that I would like to use this system frequently";
- "I found the system unnecessarily complex";
- "I thought the system was easy to use";
- "I think that I would need the support of a technical person to be able to use this system";
- "I found the various functions in this system were well integrated";
- "I thought there was too much inconsistency in this system";
- "I would imagine that most people would learn to use this system very quickly";
- "I found the system very cumbersome/complicated to use";
- "I felt very confident using the system";
- "I needed to learn a lot of things before I could get going with this system".

The rating of this statements is then computed according a given formula to obtain a global score. To resume, both quantitative and qualitative data were collected through measurements, questionnaires, observations and interviews.

5.2.1.1 Definition of Gestures for the Prototype

Before using the prototype, all participants were asked to define gestures for the available commands:

1. Start the application;
2. Add a message;
3. Review the message;
4. Edit a specific word during the reading for the review;

5. Confirm one of the five proposals made by the application after editing a word;
6. Enter a search token to find a contact;
7. List the contacts that match the search token(s);
8. Select one of the contact during the listing;
9. Send the message.

Participants were told they had no constraints about the gestures they could perform, they could do mid-air gestures, gestures on their body, or whatever they found appropriate. All participants defined the gestures and performed all tasks while seated on a chair.

5.2.2 User Study With Sighted Users

The purpose of this study was to assess the evaluation protocol and to be sure that it would be doable with visually impaired people. But as well to have a first feedback and make some improvements on the prototype based on this feedback. Information about the participants, a small analysis of the results and some general observations are presented in this section.

5.2.2.1 Participants

The evaluation was made in a first time with sighted people (n=8, 7 males & 1 female) from 26 to 42 years old ($M = 30.875, SD = 5.84166$), they owned a smartphone for 1 to 10 years ($M = 5.75, SD = 3.41216$), was all beginner concerning the use of TalkBack and nearly all of them classified themselves as expert in using the SMS application that they used nearly every day.

In a first time, it was planned to ask the participants to perform 3 sets of tasks composed of one easy task, one normal task and one hard task, for a total of 18 tasks. But after that the first participant took nearly 4 hours to complete the evaluation, only 2 sets of tasks were asked afterwards. It was also decided to stop the participant after 5 minutes if he had not completed the subtask in progress (entering the message, reviewing and editing the message or choosing the contact) in order to gain some times, this was not applied with the visually impaired.

The same smartphone was used for both the standard application and the prototype. Having no experience in using TalkBack, a small tutorial was given to the participants before they used it to complete the tasks with the standard application. This tutorial shows the participant how to navigate through the elements on the screen, how to select and confirm one element, how to navigate in a list, how to use the keyboard, how to navigate through a text and how to open the menu to change the granularity to navigate in a text. A small trial phase was given to the participant before and after he was blindfolded.

5.2.2.2 Results

This section presents some basic statistics computed on the results gathered during the sessions. First here are the mean (M), the standard deviance (SD), the minimum (Min) and the maximum (Max) of the times to perform the tasks for both systems.

Table 5.1: Time in seconds to perform the task - Set 1 - Sighted Users

System	***Easy***	***Normal***	***Hard***
Developed Application	$M = 52.625$	$M = 70.75$	$M = 106.375$
	$SD = 16.91945$	$SD = 19.6959$	$SD = 24.99678$
	$Min = 36$	$Min = 32$	$Min = 81$
	$Max = 86$	$Max = 99$	$Max = 159$
Standard Service	$M = 203.625$	$M = 506$	$M = 428.625$
	$SD = 113.58815$	$SD = 86.17258$	$SD = 70.80345$
	$Min = 119$	$Min = 411$	$Min = 380$
	$Max = 471$	$Max = 684$	$Max = 485$

Table 5.2: Time in seconds to perform the task - Set 2 - Sighted Users

System	***Easy***	***Normal***	***Hard***
Developed Application	$M = 36.71429$	$M = 89.125$	$M = 113.28571$
	$SD = 13.00915$	$SD = 16.28704$	$SD = 17.41373$
	$Min = 20$	$Min = 67$	$Min = 81$
	$Max = 63$	$Max = 110$	$Max = 137$
Standard Service	$M = 183.83333$	$M = 394.83333$	$M = 485.83333$
	$SD = 149.06833$	$SD = 80.86882$	$SD = 104.34446$
	$Min = 62$	$Min = 317$	$Min = 334$
	$Max = 386$	$Max = 522$	$Max = 597$

It can be observed that, of course, the time to perform the task rises with the difficulty of the task with both systems. The difference is slightly smaller with the standard service between the normal and the hard task which can be explained by the fact that the participant was stopped after 5 minutes if he had not completed the subtask. It is obviously the edition that takes the most time and so if the participant was already stopped when he had to only edit 1 word, he was also likely to be stopped when he had to edit 2 words.

Regarding the extremums, most of the difference can be explained with the experience that the participant had gained during the evaluation, it is natural that a participant that had performed the first set 1 after the set 2 had better performance time than one that had started with the set 1. For the standard service, the strategy used by the participant for the edition can also be decisive. Indeed, the real difficulty was to locate the word in the sentence to erase it and then add the new word with the speech recognizer triggered through the keyboard. So, to overcome that, some participants erased the whole sentence and then replace it with the new one directly and thus, they gained time this way.

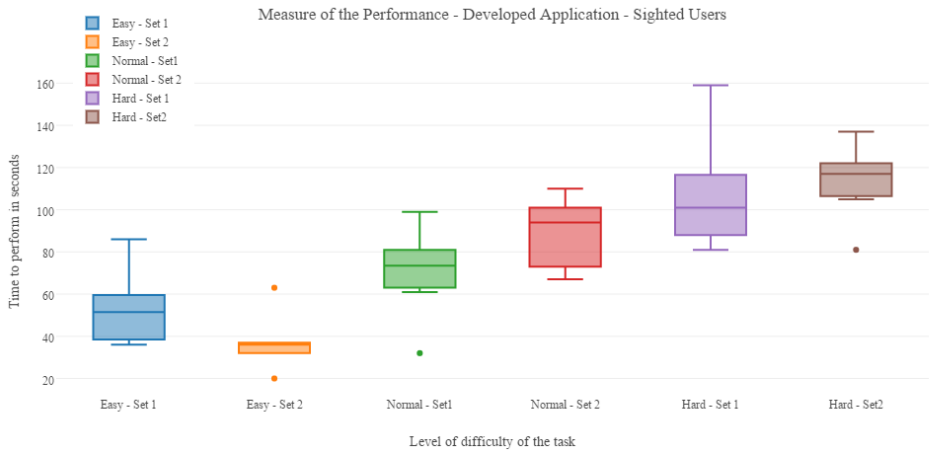

Figure 5.2: Box Plots of the time measurements of the use of the developed prototype by sighted users

The chart of the figure 5.2 confirms the fact that the time increases with the level of difficulty. It also shows that besides some extremums the values are not scattered, so it seems that the data distribution is normal. Some Shapiro-Wilk tests were performed for each task and every time the p-value is higher than 0.05 and so it cannot be concluded that the distribution deviates from normality.

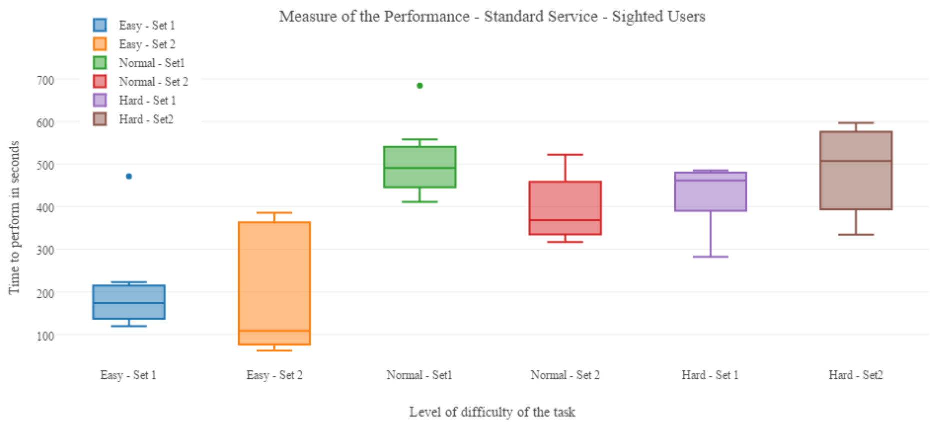

Figure 5.3: Box Plots of the time measurements of the use of their classical SMS service by sighted users

Again, the chart of the figure 5.3 visually emphasises the observation stated above. Shapiro-Wilk tests were also performed and led to the same conclusion as those stated before about the time when using the prototype.

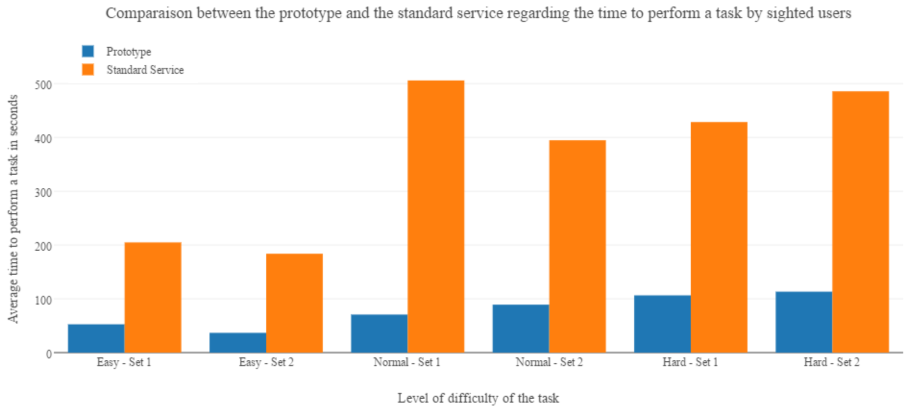

Figure 5.4: Comparison between the prototype and the standard service regarding the time to perform a task by sighted users

The chart here above (figure 5.4) shows well that the prototype was way faster than the standard service (in this case the standard Android SMS application coupled with TalkBack). The difference could have been much greater for the hard tasks if the limit of 5 minutes would not have been established.

This also reveals the fact that TalkBack is clearly not intuitive for sighted users which can be easily lost. Indeed, there are used to locate directly visually where to push and here they have to navigate through the entire interface and wait for the feedback to be sure to activate the right component. This can lead to such things as writing the message in the textfield reserved for the contact or difficulties to know where the writing cursor is located. Therefore, it also has the merit to show them how it can be frustrating to use a device that was clearly not made to be used blindfolded.

Table 5.3: Preferred Delays - Sighted Users

Delay	***1 second***	***1.5 seconds***	***2 seconds***
Reviewing	3	4	1
Listing contacts	1	5	2

As mentioned in the global proceedings, participants were asked to rate their preference over 3 different reading speeds during the selection a word in the message for an edition and during the selection of a contact from the list of contacts. Basically, the default delay of 1.5 seconds was preferred by most of the participants for both context (44% for the edition and 56% for the listing of contacts). Some choose 1 second, stipulating that with some training, they will get used to it. During these tests every participant was a bit late to choose the specific contact with the 1 second delay, so it confirms the fact that they need more practice.

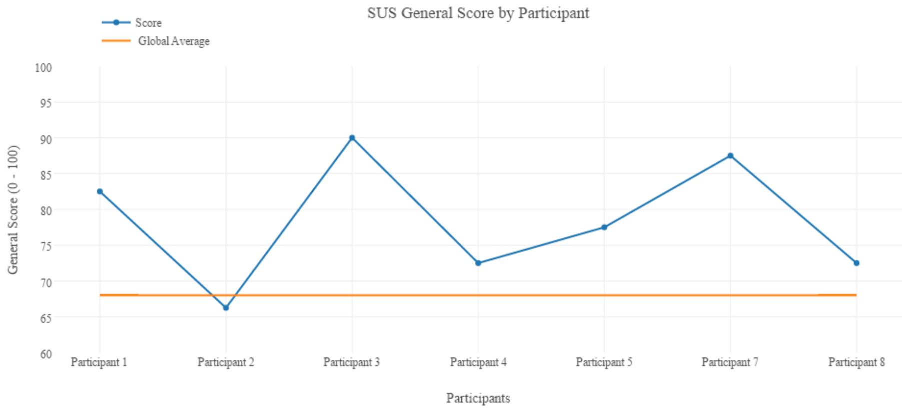

Figure 5.5: SUS general score by sighted participant

Regarding the SUS scores, based on research of Sauro [39], a SUS score above a 68 would be considered above average and anything below this value is considered below average. So it appears that 6 participants are globally satisfied by the developed application and 1 is just above this average. Participant 6 has not been taken into account since he does not rate all the statements.

5.2.2.3 Gesture Definition & Memorization

Regarding the gesture definition, the participant tended to choose more mimetic gestures. For instance, to open the application these gestures were performed:

- Mimicking the hand gesture writing gesture;
- Drawing a rectangle to represent the envelope which is usually the visual icon for an SMS application;
- Imitating the opening of a door;

Another gesture that was quite popular to edit a specific word, to confirm a proposal or a contact was to perform the stop sign as shown in the figure 5.6 on the right. Since these tasks are similar, participants were allowed to reuse the gesture.

Figure 5.6: Stop hand gesture

Regarding the memorization of these gestures, the participants were asked to repeat twice the gestures that they chose before performing the tasks. They sometimes had some issues remembering some of them but after two or three tasks, they seemed to become more comfortable.

5.2.2.4 Global Observations & Participants' Feedbacks

There were sometimes misspelling errors from the speech recognizer but globally the recognition is correct. The problem, of course, is that it requires an internet connexion to be optimal. Indeed, even in English, the offline recognizer is quite poor.

In a first time, when editing, there was no feedback of the word chosen and so the participant had no information that it was the correct word that was chosen. Thus, to rectify that, a vocal feedback was added after a few evaluations. Some participants also sometimes missed or forgot to choose one of the proposals. Again, a vocal feedback was added before the application lists the proposals to remind the participant to choose one of the five.

It also appeared that nearly none of the participant entered more than one token when searching for a contact. Even if the application told them that the matching contains more than five matches, they preferred to directly list these matches than reducing the scope of the results.

5.2.3 User Study With Unsighted Users

Given the duration to complete the tasks of 2 sets, it was decided to only use one set. Like this the evaluation lasted approximately 1 hour and 45 minutes, it was split into two sessions (one by application) of one hour each in two different days.

Again, a smartphone was provided to perform the evaluation related to the prototype. For the other condition, participants used their own smartphone and their preferred messaging application. An access to the participants' smartphone was requested before starting the tasks to collect data about their list of contacts in order to make sure that all tasks had similar difficulty.

As the number of tasks was limited to 3 and being a novel form of interaction, a training task was given before completing the tasks in the prototype. This allows the participants to get acquainted with the flow of the application and recall all the gestures they defined moments before.

This time, the evaluation was undertaken in Portuguese and so, all the messages were translated and the prototype was switched to this language.

5.2.3.1 Participants

The group of participants was composed by 9 persons (6 males & 3 females) which were recruited in a blind foundation. Their ages ranged from 24 to 56 years old ($M = 32.78, SD = 9.62$). Eight of the participants lost their vision between 14 and 31 years old and one has partial vision since birth.

Regarding mobile phone consumption habits, 6 participants stated to own a smartphone for approximately 1 year. Five persons own an Android phone and four an iPhone (only one of them use Siri). When asked in what level they consider to be using the assistive technology built in (Talkback or VoiceOver), 3 considered themselves experts, 3 proficient, 2 average and one as a beginner. In the specific case of using the messaging application, 2 considered to be experts while 4 of them considered to be proficient, 2 average users and 1 considered himself as a beginner. Nearly all participants send and receive messages every day (8 out of 9).

When asked if when walking outside, they stop answering an incoming call or an SMS or they ignore it, 6 participants stated to answer while walking.

Regarding the acceptability of performing midair gestures in any context, public or not, 4 participants stated they would do it. The other 5 participants stated that it would depend on the gestures (subtle or not) as well as the location.

5.2.3.2 Results

Table 5.4: Time in seconds to perform the task - Blind Users

System	***Easy***	***Normal***	***Hard***
Developed Application	$M = 59.66667$	$M = 93.77778$	$M = 132.11111$
	$SD = 15.49193$	$SD = 21.29424$	$SD = 23.31011$
	$Min = 39$	$Min = 68$	$Min = 101$
	$Max = 79$	$Max = 139$	$Max = 167$
Standard Service	$M = 207.33333$	$M = 255.11111$	$M = 312$
	$SD = 101.33114$	$SD = 93.46449$	$SD = 139.30721$
	$Min = 30$	$Min = 82$	$Min = 76$
	$Max = 365$	$Max = 364$	$Max = 500$

It can also be observed that two trends seem to emerge. As expected, more difficult tasks took more time to be completed and tasks were completed quicker when made with the prototype. To validate these observations, we need to look for the effects of the two independent variables (the task difficulty and the application used) on the dependent variable (the time to complete the task). But first it is necessary to assess the normality of the data.

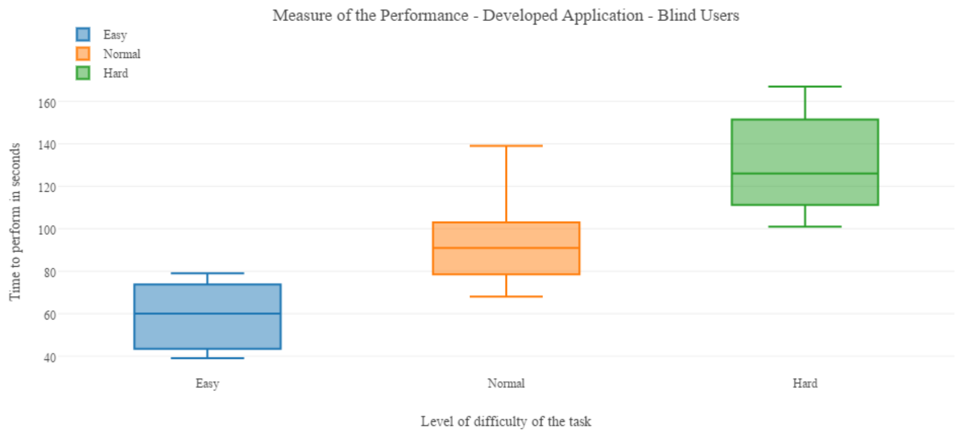

Figure 5.7: Box Plots of the time measurements of the use of the developed prototype by blind users

As shown on these box plots, the distribution of the data seems normal, some Shapiro-Wilk tests were performed and they also confirm that.

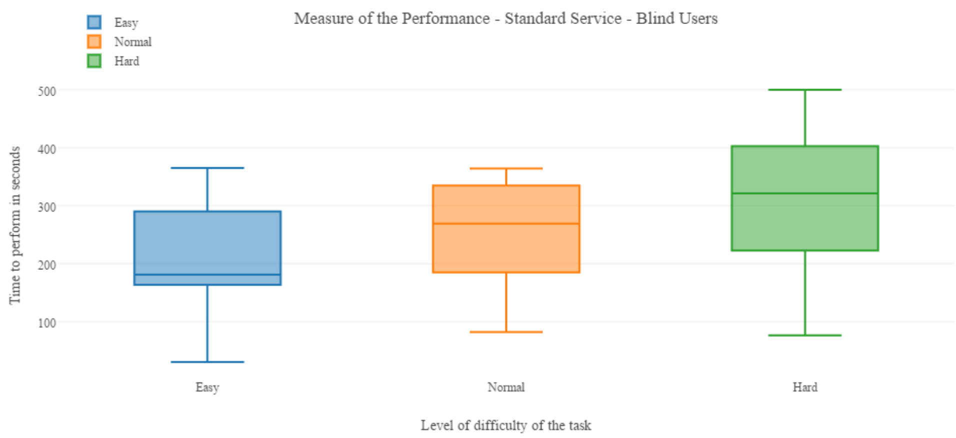

Figure 5.8: Box Plots of the time measurements of the use of their classical SMS service by blind users

The same conclusion can also be drawn regarding the normality of the time when using the standard application. Again, some Shapiro-Wilk tests were performed and they do not turn out be positive.

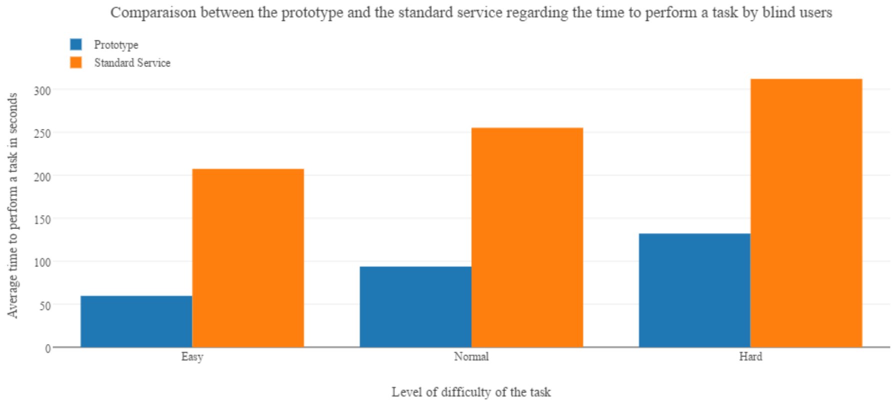

Figure 5.9: Comparison between the prototype and the standard service regarding the time to perform a task by blind users

Now that the normality is assessed, more tests were conducted on these data to know if there is a main effect between the two independent variables (the difficulty level and the use of the developed application or the standard service) that impacts the time to perform a task.

The ANOVA test was not conclusive ($F(2, 16) = 0.4103, p = 0.5247$) showing that there is no such interaction effect but simple main effects for both variables are present: the task difficulty ($F(2, 16) = 11.1232, p = 0.0016$) and the application used ($F(1, 8) = 55.6130, p < 0.001$)).

Thus, main effect pairwise comparisons were executed for both independent variables:

- *Task difficulty:* easy with normal ($F = 0.296, p = 0.0167$), easy with hard (F=0.025, p=0.0167) and normal with hard (F=0.214, p=0.0167), there is so no statistical evidence that the difficulty of the task so basically the number of editions and the difficulty to find a contact influence the time.
- *Application used:* $F = 8.4e^{-9}$ and $p = 0.05$, which confirms that there a significant effect based on the application used.

The results for the task difficulty are quite inconsistent with the observation made on the various charts. It is possible that size of the message influences the results since it varies strongly in function of the difficulty. And so, to assess if the number of editions affects the time to complete the task, it is interesting to perform this statistical analysis again on the time without counting the writing of the message.

Table 5.5: Time in seconds to perform the task without the writing of the message - Blind Users

System	***Easy***	***Normal***	***Hard***
Developed Application	$M = 34.78$	$M = 83.22$	$M = 118.89$
	$SD = 10.77$	$SD = 21.54$	$SD = 21.03$
	$Min = 23$	$Min = 58$	$Min = 109$
	$Max = 51$	$Max = 129$	$Max = 147$
Standard Service	$M = 39$	$M = 146.11$	$M = 238.11$
	$SD = 21.54$	$SD = 66.08$	$SD = 112.74$
	$Min = 19$	$Min = 64$	$Min = 67$
	$Max = 84$	$Max = 263$	$Max = 389$

It appears that now the difference between the two systems is smaller but the chart of the figure 5.10 still confirms the fact that the developed prototype is faster. And again the two same trends then before seems to appear, more difficult tasks require more time to be completed and the task were achieved quicker with the prototype. Firstly, the normality of the data was checked and the results that the data follows a normal distribution except the group of data of the easy task on the standard application.

Even with this violation of normality, a parametric two-way repeated measures ANOVA test was still performed thanks to its robustness against this kind of violation. The results show that there is a significant interaction ($F(2, 16) = 8.024, p = 0.004$) between the two independent variables (the task difficulty and the system used). Now, the difference between each level of each factor needs to be determined.

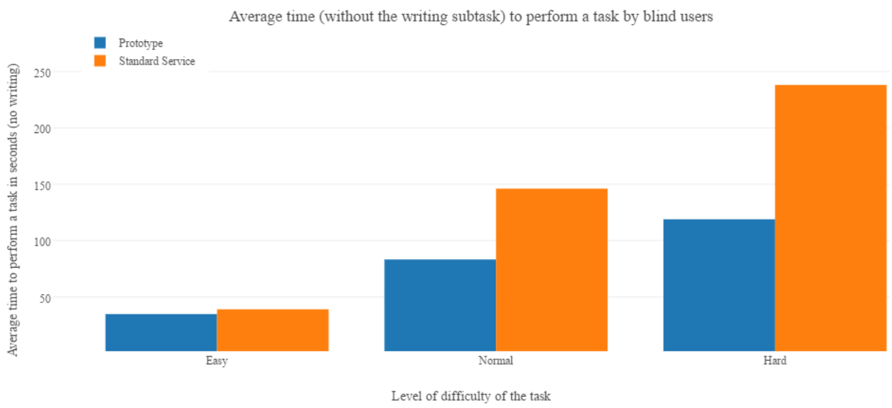

Figure 5.10: Comparison between the prototype and the standard service regarding the time to perform a task (with no writing task) by blind users

The first factor to be analysed was ***the application used***. A simple main effect test was computed to observe the differences between the two applications for each level of task difficulty. Results showed that, for the easy task, any significant difference could not be found between the two applications ($p = 0.664$). However, statistical differences were found in the normal ($p = 0.026$) and hard ($p = 0.012$) tasks. Therefore, it can be concluded that the prototype allowed the study participants to complete the tasks quicker than the standard SMS application in the tasks that required message editing, so, the normal and hard tasks. Therefore, the prototype supports faster message correction than the standard solution. This is the result of the careful design of the message definition and of the correction mechanism. In the task that simply required text input and the selection of a contact in the beginning of the list of contacts, there were no differences between the applications, even though, on average, the prototype was still more than 4 seconds quicker than the standard application.

The second factor analysed was ***the task difficulty***. Another simple main effect test was performed to assess the differences between task difficulty within each application. Observing the pairwise comparisons, the results show a significant difference for easy and normal tasks ($p < 0.001$), easy and hard tasks ($p < 0.001$) and normal and hard tasks ($p = 0.001$) for the prototype. The same results occurred in the standard application: there is also a significant difference between easy and normal ($p = 0.001$) tasks, easy and hard tasks ($p < 0.001$) and normal and hard tasks ($p = 0.012$). To conclude with the task difficulty, more difficult tasks take more time to be completed, independently of the application used, which is an expected result.

Table 5.6: Preferred Delays - Blind Users

Delay	***1 second***	***1.5 seconds***	***2 seconds***
Reviewing	5	3	1
Listing contacts	4	4	1

Regarding the reading speed preference for the edition and listing of contacts, findings show that despite 22% of the participants fail to select the right word while using the fastest speed, over 55% preferred this option. Again as previously with sighted users, those who failed stated they could do it with more training. On the other hand, when selecting a contact from the list, there was not a preferred reading speed choice. Four participants preferred the fastest speed and four others preferred the middle one. However, 33% of the participants failed when selecting a contact in the fastest speed mode. One possible solution to offer this speed and in the same time counter the mistake will be to use a correction algorithm that would adapt the selection in function of the users' reaction time.

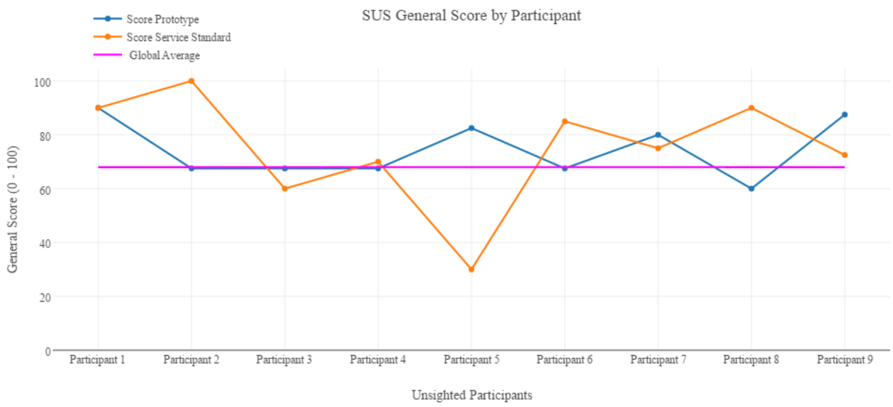

Figure 5.11: SUS general score by unsighted participant

Both applications rated high on the SUS scale as shown on the figure 5.11. The prototype averaged 74.44 points, while the standard SMS applications rated, on average, 74.72 points. And so according to the SUS scoring system [39], results higher than 68 are above average and so considered usable.

Some higher scores for the standard can be explained by the fact that the participants are used to this solution and through this experience, they think that this is the best and simple solution. Therefore, the participants feel more comfortable and confident because of this past experience with this system.

5.2.3.3 Gesture Definition & Memorization

The participants were also asked to define a gesture for each command. And so in total, 81 gestures were gathered. This time, few of theses gestures were repeated among the participants. However a trend can be observed for 4 commands:

- *Adding text*: Movement similar to writing with the fingers and performed on the leg, this was chosen by the 2 participants;
- *Selecting a proposal.* The index finger touch which was chosen by 3 participants;
- *Selecting a contact during the list review.* Again the index finger touch was chosen by the 3 same participants;
- *Sending the message.* Swipe the hand to the right, this was chosen by 3 participants.

So it can be noticed that a gesture was used twice by some participants to perform similar actions, it was also the case for the other participants which also claimed that it is an easy way to decrease the number of gesture to memorize.

Regarding these declarations, it is more surprising that only 1 participant also used the same gesture for editing a specific word during the reading, which is a similar action to the proposal and contact's selection.

Regarding privacy, 3 out of 9 participants defined more subtle gestures and most of them were performed beside the participant's pockets or on an available surface.

Additionally, it was observed that several gestures meant different actions depending on the participant:

- *Thumbs up*: One participant considered this gesture to start writing the message while another participant chose this gesture to send the SMS.
- *Swipe up with two fingers.* This gesture was defined to review the message but also to listen to the contacts list.
- *Drawing of the letter C*: One participant chose this gesture to enter a search token to find a contact while another made the same gesture to listen to the contacts list that matched with the token;
- *Index finger double touch.* This gesture was defined to start writing the message and also to listen to the list of contacts that matched the search token;
- *Point with index finger.* One participant chose this gesture to open the application while another chose it to start writing the message.

These multiple interpretations combined with the fact that only a few gestures were found for the same action across the participants. Even though the sample of recruited participants is small, this diversity of proposed gestures leads to the conclusion that users need to be able to define their own gestures and thus the interface should support this customization.

While performing the tasks on the prototype, 26 gesture related errors were registered. On average 0.96 error occurred per participant ($SD = 1.20$). The following mistakes are considered as errors:

1. Forgetting what gesture to do to trigger an action;
2. Performing the wrong gesture;
3. Selecting a wrong list item due to taking too long time to recall the gesture.

5.2.3.4 Participants' Feedbacks & Publication of the Results

5 participants stated that *"this application is much easier than the one that already exist for the blind"* and 2 other participants said it to be *"very functional upon learning the gestures."*. Moreover, 8 participants stated that the *"keyboard is too small which can lead to errors while writing the message"* when discussing the current standard applications.

However, two participants raised concerns about using an online voice recognizer, stating the limitations of being required to be always connected to the internet because of the lack of performance of current offline versions. This was also mentioned during the evaluations with sighted users.

The results presented here were the basis for a scientific paper which can be read in the appendix B on page 97. This paper was submitted in May 2016 to the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2016).

Conclusion

Overview

To summarize, this thesis first presents some global key concepts that were used to create the proposed architecture in chapter 4 and then 5. Among these concepts, multimodal interfaces as well as the principle of context adaptation are briefly explained. Behavioural information and special concerns of the visually impaired are then detailed in chapter 2 along with a list of possible modalities and some interesting technologies regarding their disability.

The main contribution of this thesis is the case study presented in chapter 5. It consists in an SMS application based on the idea of accessing functionalities directly through short-cuts as presented in section 4.1. Indeed, the user can directly perform some actions (write or edit a message and select a contact) through gestures and speech. This allows to make the interaction with the smartphone hand-free where standard application usually required both hands, an important difference when considering that the targeted users may have to hold a cane or a guide dog.

User studies were performed to assess the suitability and the performance of the prototype in comparison with standard applications. The first one with sighted users allowed to make sure that the evaluation protocol was doable later with visually impaired users and to have a first feedback which led to some little improvements in the prototype as explained in section 5.2.2.4. The second one with the visually impaired participants showed that the proposed solution had a better performance when text editing was required in comparison with current available text messaging applications. The participants also gave positive feedback regarding the use of gestures but some stated that it must stay subtle in order to be socially acceptable. Moreover, regarding the diversity of the proposed gestures, the experiment emphasizes the usefulness of a customized and adaptive system in order to be usable and efficient for most users.

Finally, the results gathered during the evaluations with unsighted users and presented in section 5.2.3 were the basis for a scientific paper which can be read in the appendix B on page 97. This paper was submitted in May 2016 to the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2016).

Critical Outlook

It could have been interesting to meet the visually impaired users before developing the prototype to have a first feedback from them regarding the functioning and the proposed interaction method. But also, to already observe them with their classical SMS application which could have been inspirational.

Moreover, the biggest challenge is still to extend the proposed interaction method to the whole operating system. This can perhaps be achieved in the form of an accessibility service as presented in section 4.2, that would allow all applications to benefit from it.

The capability to integrate a fully functional gesture recognizer which takes into account the feedback obtained during the experiment (as subtle gestures and the possibility to customize gestures) also raises challenges. However, some technologies as the Google Soli chip presented in section 2.2.4.1 are very promising.

Perspectives and Future Works

Adding a module to capture the context as presented in section 4.1.1 could really improve the usability of the prototype by allowing to choose the right modality or to adjust it correctly in function of the user's context. For instance, to reduce the delay of reading or improve the speech rate of the text-to-speech when noticing that the user is quite comfortable with the system if he nearly makes no mistakes. It could be possible to use frameworks that already exist to capture some components of the context. Indeed, the Aware Framework [3] logs and shares mobile context data (hardware, software and human-based data) and then, AWARE plugins analyse the data and infer understandable information.

Coupled with the context adaptation, the addition of new modalities, as the ones presented in section 2.2, can also be considered to improve even more the usability. Indeed, regarding the context, Braille could be a good alternative to input text. Vibration could also be used to replace orally feedback once the user is used to the application and so, that this feedback does not need to carry too much information any more.

[3] http://www.awareframework.com/

Bibliography

[1] Tousif Ahmed, Roberto Hoyle, Kay Connelly, David Crandall, and Apu Kapadia. Privacy Concerns and Behaviors of People with Visual Impairments. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3523–3532. ACM, 2015.

[2] Damien Appert, Damien Camors, Jean-Baptiste Durand, and Christophe Jouffrais. Assistance tactile à la localisation de cibles périphériques pour des personnes à vision tubulaire. *IHM'15 - Proceedings of the 27nd French-speaking conference on Human-computer interaction*, 2015.

[3] Damien Appert, Christophe Jouffrais, and Emmanuel Dubois. Interactions non-visuelles pour faciliter le rendu d'une séquence d'informations. In *27ème conférence francophone sur l'Interaction Homme-Machine.*, page w11. ACM, 2015.

[4] Shiri Azenkot and Nicole B. Lee. Exploring the use of speech input by blind people on mobile devices. *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '13*, pages 1–8, 2013.

[5] Shiri Azenkot, Sanjana Prasain, Alan Borning, Emily Fortuna, Richard E. Ladner, and Jacob O. Wobbrock. Enhancing Independence and Safety for Blind and Deaf-Blind Public Transit Riders. *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*, pages 3247–3256, 2011.

[6] Richard A Bolt. *Put-that-there: Voice and gesture at the graphics interface*, volume 14. ACM, 1980.

[7] Monica Bordegoni, Giorgio Faconti, Steven Feiner, Mark T Maybury, Thomas Rist, Salvatore Ruggieri, Panos Trahanias, and Michael Wilson. A standard reference model for intelligent multimedia presentation systems. *Computer standards & interfaces*, 18(6):477–496, 1997.

[8] Erin L. Brady, Yu Zhong, Meredith Ringel Morris, and Jeffrey P. Bigham. Investigating the appropriateness of social network question asking as a resource for blind users. *Proc. CSCW 2013*, page 1225, 2013.

[9] Eoin Brazil and Mikael Fernstöm. Auditory icons. *The Sonification Handbook*, pages 301–324, 2011.

[10] John Brooke. SUS - A quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

[11] Gaëlle Calvary, Joëlle Coutaz, David Thevenin, Quentin Limbourg, Laurent Bouillon, and Jean Vanderdonckt. A Unifying Reference Framework for multi-target user interfaces. *Interacting with Computers*, 15:289–308, 2003.

[12] George Collins and David Sisk. API Economy: From Systems to Business Services. `http://www2.deloitte.com/content/dam/Deloitte/us/Documents/financial-services/us-fsi-api-economy.pdf`, 2015. (Accessed on 07/11/2016).

[13] World Wide Web Consortium. Accessibility, Usability, and Inclusion. `https://www.w3.org/WAI/intro/usable`, May 2016. (Accessed on 25/07/2016).

[14] David Costa and Carlos Duarte. Adapting multimodal fission to users abilities. In *International Conference on Universal Access in Human-Computer Interaction*, pages 347–356. Springer, 2011.

[15] David Costa and Carlos Duarte. From one to many users and contexts: A classifier for hand and arm gestures. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pages 115–120. ACM, 2015.

[16] Joëlle Coutaz, Laurence Nigay, Daniel Salber, Ann Blandford, Jon May, and Richard M Young. Four easy pieces for assessing the usability of multimodal interaction: the CARE properties. *InterAct*, (June):1–7, 1995.

[17] Joëlle Coutaz and Gaëtan Rey. Foundations for a theory of contextors. In *Computer-Aided Design of User Interfaces III*, pages 13–33. Springer, 2002.

[18] Charalampos Doukas and Fabio Antonelli. COMPOSE: Building smart & context-aware mobile applications utilizing IoT technologies. *Global Information Infrastructure Symposium, GIIS 2013*, 2013.

[19] Bruno Dumas, Denis Lalanne, and Sharon Oviatt. Multimodal interfaces: A survey of principles, models and frameworks. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5440 LNCS:3–26, 2009.

[20] Mary Ellen Foster. State of the art review: Multimodal fission. *COMIC project Deliverable*, 6(09), 2002.

[21] Brian Frey, Caleb Southern, and Mario Romero. Brailletouch: mobile texting for the visually impaired. In *International Conference on Universal Access in Human-Computer Interaction*, pages 19–25. Springer, 2011.

[22] Paul Green, Lisa Wei-Haas, et al. *The Wizard of Oz: A Tool for Rapid Development of User Interfaces. Final Report.* University of Michigan, Transportation Research Institute, 1985.

[23] Gerd Herzog and Norbert Reithinger. The smartkom architecture: A framework for multimodal dialogue systems. In *SmartKom: Foundations of Multimodal Dialogue Systems*, pages 55–70. Springer, 2006.

[24] Frank Honold, Felix Schüssel, and Michael Weber. Adaptive probabilistic fission for multimodal systems. In *Proceedings of the 24th Australian Computer-Human Interaction Conference*, pages 222–231. ACM, 2012.

[25] Shaun K. Kane, Chandrika Jayant, Jacob O. Wobbrock, and Richard E. Ladner. Freedom to roam: a study of mobile device adoption and accessibility for people with visual and motor disabilities. In *Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility*, pages 115–122. ACM, 2009.

[26] Shaun K. Kane, Meredith Ringel Morris, Annuska Z. Perkins, Daniel Wigdor, Richard E. Ladner, and Jacob O. Wobbrock. Access Overlays: Improving Non-Visual Access to Large Touch Screens for Blind Users. *Proceedings of the 24th annual ACM symposium on User interface software and technology - UIST '11*, page 273, 2011.

[27] Shaun K. Kane, Jacob O. Wobbrock, and Richard E. Ladner. Usable gestures for blind people: understanding preference and performance. *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*, pages 413–422, 2011.

[28] Frédéric Landragin. Physical, semantic and pragmatic levels for multimodal fusion and fission. In *Proceedings of the Seventh International Workshop on Computational Semantics (IWCS-7)*, pages 346–350. Universitätsverlag Tilburg, 2007.

[29] David Mcgookin and Stephen Brewster. Earcons. *The Sonification Handbook*, pages 339–361, 2011.

[30] Linda Mohand Oussaïd, Yamine Ait Ameur, and Mohamed Ahmed-Nacer. A generic formal model for fission of modalities in output multi-modal interactive systems. In *International Workshop on Verification and Evaluation of Computer and Communication Systems*, pages 1–12, 2009.

[31] Linda Mohand Oussaïd, Idir Aït Sadoune, and Yamine Aït Ameur. A formal model for output multi-modal HCI . *Computing*, 97(7):713–740, 2014.

[32] Laurence Nigay. *Conception et modélisation logicielles des systèmes interactifs: application aux interfaces multimodales*. PhD thesis, Université Joseph-Fourier-Grenoble I, 1994.

[33] Laurence Nigay and Joëlle Coutaz. A design space for multimodal systems: concurrent processing and data fusion. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '93*, (January):172–178, 1993.

[34] World Health Organization. Visual Impairment and Blindness. `http://www.who.int/mediacentre/factsheets/fs282/en/`, August 2014. (Accessed on 04/04/2016).

[35] Martin Pielot, Benjamin Poppinga, and Susanne Boll. Pocketnavigator: vibro-tactile waypoint navigation for everyday mobile devices. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, pages 423–426. ACM, 2010.

[36] Julie Rico and Stephen Brewster. Usable gestures for mobile interfaces: evaluating social acceptability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 887–896. ACM, 2010.

[37] Cyril Rousseau, Yacine Bellik, and Frédéric Vernier. Multimodal output specification / simulation platform. *Proceedings of the 7th international conference on Multimodal interfaces - ICMI '05*, page 84, 2005.

[38] Cyril Rousseau, Yacine Bellik, and Frédéric Vernier. Wwht: Un modèle conceptuel pour la présentation multimodale d'information. In *Proceedings of the 17th Conference on l'Interaction Homme-Machine*, pages 59–66. ACM, 2005.

[39] Jeff Sauro. Measuring Usability with the System Usability Scale (SUS). `http://www.measuringu.com/sus.php`, February 2011. (Accessed on 20/07/2016).

[40] Kristen Shinohara and Jacob O. Wobbrock. In the shadow of misperception: assistive technology use and social interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 705–714. ACM, 2011.

[41] Caleb Southern, James Clawson, Brian Frey, Gregory Abowd, and Mario Romero. An evaluation of brailletouch: mobile touchscreen text entry for the visually impaired. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services*, pages 317–326. ACM, 2012.

[42] Google's Advanced Technology and Projects (ATAP). Project Soli. `https://atap.google.com/soli/`. (Accessed on 30/06/2016).

[43] David Thevenin and Joëlle Coutaz. Plasticity of user interfaces: Framework and research agenda. In *Proceedings of INTERACT*, volume 99, pages 110–117, 1999.

[44] Shaomei Wu and Lada A Adamic. Visually impaired users on an online social network. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3133–3142. ACM, 2014.

[45] Hanlu Ye, Meethu Malu, Uran Oh, and Leah Findlater. Current and future mobile and wearable device use by people with visual impairments. *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pages 3123–3132, 2014.

[46] Atef Zaguia, Ahmad Wahbi, Chakib Tadj, and Amar Ramdane-Cherif. Multimodal fission for interaction architecture. 2013.

Appendix

Appendix A

User Studies Preparation

This appendix presents the entire protocol that was followed during the user studies presented in section 5.2.

User Studies Preparation

1 Organizing questions

1.1 Details of each question

Each question should be characterized by:

- Question number,
- Question category :
  - Gesture, relative to question used to know more about the gestures that make sense to perform a specific action,
  - Performance, relative to question used to evaluate the performance of the prototype,
  - Preference, relative to question used to know what the user prefers between different configurations.
- Question text, fully describing the aim of the question,
- Testing scenario, describing a scenario leading to the answer,
- Data format, describing the data that should be gathered during the user trials, when implementing the scenario, that will allow drawing conclusions about the question posed.

For each question, the following table should be filled.

Number:	**Category:**
Question:	
Testing Scenario:	
Data to gather:	

2 Scripts

Number: Gest1	**Category:** Gesture

Question: Which gesture are performed by the user to accomplish the different action?

Testing Scenario:
Let the user choose what gesture makes most sense to him regarding a specific action. The actions are the following:

- Adding text
- Reading the text
- Editing a specific word
- Confirming one of the proposal during the edition
- Entering a token to search a contact
- Listing the contacts that match the actual search token
- Confirming one of the contact during the listing

Data to gather after the trial (video recording):
Qualitative – Observations of participants' gestures

Number: Perf1	**Category:** Performance

Question: How long does it takes to perform the different actions with the prototype? The result of this question can be used to compare it with the use of a traditional SMS' service (Perf2).

Testing Scenario:
Let the user go through the entire process to send a SMS (writing the text, checking the content of the message and editing it if needed, choose a contact). Different scenarios (defined it the next section) can be considered:

- Happy scenario: No edition of the text, contact found with a small research,
- Normal scenario: Few edition to make, contact found with a relatively small search,
- Worst scenario: A lot of edition needed, contact found with a relatively deep search (wrong search token can also be considered).

Data to gather during the trial :
Quantitative – Time to accomplish the action

Number: Perf2	**Category:** Performance

Question: How long does it takes to perform the different actions with their traditional SMS' service supported by the Android accessibility service Talk Back? The result of this question can be used to compare it with the use of the prototype (Perf1).

Testing Scenario:
Let the user go through the entire process to send a SMS (writing the text, checking the content of the message and editing it if needed, choose a contact). Different scenarios can be considered:

- Happy scenario: No edition of the text, contact found with a small research,
- Normal scenario: Few edition to make, contact found with a relatively small search,
- Worst scenario: A lot of edition needed, contact found with a relatively deep search.

Data to gather during the trial :
Quantitative – Time to accomplish the action

Number: Pref1	**Category:** Preference
Question: Which delay between two words spoken by the TTS is the best regarding the user?	
Testing Scenario: Let the user read the content of the SMS with different delay (1s, 1.5s, 2s) between the words and maybe trying to edit a specific word. Same manipulation with the reading of contacts.	
Data to gather during the trial: - Which delay do you prefer?	

PARTICIPANT	**Pref1**	
USER_	**Response for the SMS content:** **Response for the listing of the contacts:**	

3 Scenarios

3.1 Perf1

3.1.1 Happy Scenarios

3.1.1.1 First Scenario – PERF1_H1

- **Instruction for the user:**

1) Enter the following message "Hello, this a test for a gesture SMS service.” (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered
3) Send that message to "Arthur"

- **What is more or less expect from the user:**

1) Make the gesture to add text,
2) Wait for the speech recognizer’s signal,
3) Say the following message: “Hello, this a test for a gesture SMS service.”,
4) Make the gesture to read content,
5) Wait for the end of the reading,
6) Make the gesture to search contacts,
7) Wait for the speech recognizer’s signal,
8) Say the letter “A” or word beginning with “A”,
9) Make the gesture to list the contact,
10) When the second contact “Arthur” is read, make the gesture to confirm a contact.

3.1.1.2 Second Scenario – PERF1_H2

- **Instruction for the user:**

1) Enter the following message " Hello, can you call me back as soon as possible?” (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered

3) Send that message to "Veronica"
- **What is more or less expect from the user:**
1) Make the gesture to add text,
2) Wait for the speech recognizer's signal,
3) Say the following message: "Hello, I'm at the airport, can you come to pick me up.",
4) Make the gesture to read content,
5) Wait for the end of the reading,
6) Make the gesture to search contacts,
7) Wait for the speech recognizer's signal,
8) Say the letter "V" or word that beginning with "V",
9) Make the gesture to list the contact,
10) When the first contact "Veronica" is read, make the gesture to confirm a contact.

3.1.1.3 Third Scenario – PERF1_H3

- **Instruction for the user:**
1) Enter the following message "I need help, I don't know where I am." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered
3) Send that message to "Kris"
- **What is more or less expect from the user:**
1) Make the gesture to add text,
2) Wait for the speech recognizer's signal,
3) Say the following message: "I need help, I don't know where I am.",
4) Make the gesture to read content,
5) Wait for the end of the reading,
6) Make the gesture to search contacts,
7) Wait for the speech recognizer's signal,
8) Say the letter "K" or word that beginning with "K",
9) Make the gesture to list the contact,
10) When the third contact "Kris" is read, make the gesture to confirm a contact.

3.1.2 Normal Scenarios

3.1.2.1 First Scenario – PERF1_N1

- **Instruction for the user:**
1) Enter the following message "I'm waiting for you at the entrance of the shop." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "waiting" (or the word that was recognized instead) with "looking"
3) Send that message to "Victor"
- **What is more or less expect from the user:**
1) Make the gesture to add text,
2) Wait for the speech recognizer's signal,
3) Say the following message: "I'm waiting for you at the entrance of the shop.",
4) Make the gesture to read content,
5) Wait for the word "airport" (or the word that was recognized instead) to be read,

6) Make the gesture to edit,
7) Wait for the speech recognizer's signal,
8) Say the following message: "bus station",
9) After the first proposal, make the gesture to confirm a proposal,
10) Make the gesture to search contacts,
11) Wait for the speech recognizer's signal,
12) Say the letter "V" or word that beginning with "V",
13) Make the gesture to search contacts,
14) Wait for the speech recognizer's signal,
15) Say the letter "I" or word that beginning with "I",
16) Make the gesture to list the contact,
17) When the first contact "Victor" is read, make the gesture to confirm a contact.

3.1.2.2 Second Scenario – PERF1_N2

- **Instruction for the user:**

1) Enter the following message "When do you want us to meet you?" (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "When" (or the word that was recognized instead) with "Where"
3) Send that message to "Arturo"

- **What is more or less expect from the user:**

1) Make the gesture to add text,
2) Wait for the speech recognizer's signal,
3) Say the following message: "When do you want us to meet you?",
4) Make the gesture to read content,
5) Wait for the word "When" (or the word that was recognized instead) to be read,
6) Make the gesture to edit,
7) Wait for the speech recognizer's signal,
8) Say the word "Where",
9) After the first proposal, make the gesture to confirm a proposal,
10) Make the gesture to search contacts,
11) Wait for the speech recognizer's signal,
12) Say the letter "A" or word that beginning with "A",
13) Make the gesture to search contacts,
14) Wait for the speech recognizer's signal,
15) Say the letter "R" or word that beginning with "R",
16) Make the gesture to list the contact,
17) When the second contact "Arturo" is read, make the gesture to confirm a contact.

3.1.2.3 Third Scenario – PERF1_N3

- **Instruction for the user:**

1) Enter the following message "Do you prefer to meet me today or tomorrow?" (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "meet" (or the word that was recognized instead) with "see"
3) Send that message to "Patrick"

- **What is more or less expect from the user:**

1) Make the gesture to add text,
2) Wait for the speech recognizer's signal,
3) Say the following message: "Do you prefer to meet me today or tomorrow?",
4) Make the gesture to read content,
5) Wait for the word "help" (or the word that was recognized instead) to be read,
6) Make the gesture to edit,
7) Wait for the speech recognizer's signal,
8) Say the following message: "see",
9) After the first proposal, make the gesture to confirm a proposal,
10) Make the gesture to search contacts,
11) Wait for the speech recognizer's signal,
12) Say the letter "P" or word that beginning with "P",
13) Make the gesture to search contacts,
14) Wait for the speech recognizer's signal,
15) Say the letter "A" or word that beginning with "A",
16) Make the gesture to list the contact,
17) When the third contact "Patrick" is read, make the gesture to confirm a contact.

3.1.3 Worst Scenarios

3.1.3.1 First Scenario – PERF1_W1

- **Instruction for the user:**

1) Enter the following message "Do you know where Brian is?" (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "Do" (or the word that was recognized instead) with "Can"
3) Review the message again and replace "know" (or the word that was recognized instead) with "tell me"
4) Send that message to "Patricia"

- **What is more or less expect from the user:**

1) Make the gesture to add text,
2) Wait for the speech recognizer's signal,
3) Say the following message: "Do you know where Brian is?",
4) Make the gesture to read content,
5) Wait for the word "Do" (or the word that was recognized instead) to be read,
6) Make the gesture to edit,
7) Wait for the speech recognizer's signal,
8) Say the word "Can",
9) After the first proposal, make the gesture to confirm a proposal,
10) Make the gesture to read content,
11) Wait for the word "know" (or the word that was recognized instead) to be read,
12) Make the gesture to edit,
13) Wait for the speech recognizer's signal,
14) Say the word "tell me",
15) After the second proposal, make the gesture to confirm a proposal,
16) Make the gesture to search contacts,
17) Wait for the speech recognizer's signal,

18) Say the letter “P” or word beginning with “P”,
19) Make the gesture to search contacts,
20) Wait for the speech recognizer’s signal,
21) Say the letter “A” or word that beginning with “A”,
22) Make the gesture to search contacts,
23) Wait for the speech recognizer’s signal,
24) Say the letter “T” or word that beginning with “T”,
25) Make the gesture to list the contact,
26) When the second contact “Patricia” is read, make the gesture to confirm a contact.

3.1.3.2 Second Scenario – PERF1_W2

- **Instruction for the user:**

1) Enter the following message "The bus is late, I will be there in forty minutes.” (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace “bus” (or the word that was recognized instead) with “train”
3) Review the message again and replace “forty” (or the word that was recognized instead) with “fifty”
4) Send that message to "Brook"

- **What is more or less expect from the user:**

1) Make the gesture to add text,
2) Wait for the speech recognizer’s signal,
3) Say the following message: “The bus is late, I will be there in forty minutes.”,
4) Make the gesture to read content,
5) Wait for the word “bus” (or the word that was recognized instead) to be read,
6) Make the gesture to edit,
7) Wait for the speech recognizer’s signal,
8) Say the following message: “train”,
9) After the first proposal, make the gesture to confirm a proposal,
10) Make the gesture to read content,
11) Wait for the word “thirty” (or the word that was recognized instead) to be read,
12) Make the gesture to edit,
13) Wait for the speech recognizer’s signal,
14) Say the following message: “fifty”,
15) After the third proposal, make the gesture to confirm a proposal,
16) Make the gesture to search contacts,
17) Wait for the speech recognizer’s signal,
18) Say the letter “B” or word that beginning with “B”,
19) Make the gesture to search contacts,
20) Wait for the speech recognizer’s signal,
21) Say the letter “R” or word that beginning with “R”,
22) Make the gesture to search contacts,
23) Wait for the speech recognizer’s signal,
24) Say the letter “O” or word that beginning with “O”,
25) Make the gesture to list the contact,
26) When the first contact “Victor” is read, make the gesture to confirm a contact.

3.1.3.3 Third Scenario – PERF1_W3

- **Instruction for the user:**

1) Enter the following message "Hello, I'm at the airport, can you come to pick me up." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "airport" (or the word that was recognized instead) with "bus station"
3) Review the message again and replace "am" (or the word that was recognized instead) with "just arrive"
4) Send that message to "Leon"

- **What is more or less expect from the user:**

1) Make the gesture to add text,
2) Wait for the speech recognizer's signal,
3) Say the following message: "Hello, I am at the airport, can you come to pick me up.",
4) Make the gesture to read content,
5) Wait for the word "airport" (or the word that was recognized instead) to be read,
6) Make the gesture to edit,
7) Wait for the speech recognizer's signal,
8) Say the following message: "bus station",
9) After the first proposal, make the gesture to confirm a proposal,
10) Make the gesture to read content,
11) Wait for the word "am" (or the word that was recognized instead) to be read,
12) Make the gesture to edit,
13) Wait for the speech recognizer's signal,
14) Say the following message: "just arrive",
15) After the second proposal, make the gesture to confirm a proposal,
16) Make the gesture to search contacts,
17) Wait for the speech recognizer's signal,
18) Say the letter "L" or word beginning with "L",
19) Make the gesture to search contacts,
20) Wait for the speech recognizer's signal,
21) Say the letter "E" or word that beginning with "E",
22) Make the gesture to search contacts,
23) Wait for the speech recognizer's signal,
24) Say the letter "O" or word that beginning with "O",
25) Make the gesture to list the contact,
26) When the third contact "Leon" is read, make the gesture to confirm a contact.

3.2 Perf2

The user will have to use Talk Back, here are the main functionalities offered by this accessibility service:

- Tap on the screen to have the detail of an element, swipe to the right for the next element and to the left for the previous element.
- To activate a component, just double tap.
- Scrolling is enabled with two fingers.

3.2.1 Happy Scenarios

3.2.1.1 First Scenario – PERF2_H1

1) Enter the following message "Hello, this a test for a gesture SMS service." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered
3) Send that message to "Arthur"

3.2.1.2 Second Scenario – PERF2_H2

1) Enter the following message "Hello, can you call me back as soon as possible?" (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered
3) Send that message to "William"

3.2.1.3 Third Scenario – PERF2_H3

1) Enter the following message "I need help, I don't know where I am." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered
3) Send that message to "Kris"

3.2.2 Normal Scenarios

3.2.2.1 First Scenario – PERF2_N1

1) Enter the following message "I'm waiting for you at the entrance of the shop." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "waiting" (or the word that was recognized instead) with "looking"
3) Send that message to "Victor"

3.2.2.2 Second Scenario – PERF2_N2

1) Enter the following message "When do you want us to meet you?" (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "When" (or the word that was recognized instead) with "Where"
3) Send that message to "Donald"

3.2.2.3 Third Scenario – PERF2_N3

1) Enter the following message "Do you prefer to meet me today or tomorrow?" (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "meet" (or the word that was recognized instead) with "see"
3) Send that message to "Pascal"

3.2.3 Worst Scenarios

3.2.3.1 First Scenario – PERF2_W1

1) Enter the following message "Do you know where Brian is?" (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "Do" (or the word that was recognized instead) with "Can"
3) Review the message again and replace "know" (or the word that was recognized instead) with "tell me"
4) Send that message to "Thomas"

3.2.3.2 Second Scenario – PERF2_W2

1) Enter the following message "The bus is late, I will be there in forty minutes." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "bus" (or the word that was recognized instead) with "train"
3) Review the message again and replace "forty" (or the word that was recognized instead) with "fifty"
4) Send that message to "Leo"

3.2.3.3 Third Scenario – PERF2_W3

1) Enter the following message "Hello, I'm at the airport, can you come to pick me up." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Review the message you just entered and replace "airport" (or the word that was recognized instead) with "bus station"
3) Review the message again and replace "am" (or the word that was recognized instead) with "just arrive"
4) Send that message to "Patricia"

3.3 Pref1

3.3.1 Reading content

3.3.1.1 1 second delay – PREF1_R1

Prerequisite: Set the delay between the reading of two words to 1 second

1) Review the following message "Hello, this a test for a gesture SMS service." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
2) Replace "gesture" with "talking"

3.3.1.2 1.5 second delay – PREF1_R1.5

Prerequisite: Set the delay between the reading of two words to 1.5 second

3) Review the following message "Hello, this a test for a gesture SMS service." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
4) Replace "gesture" with "talking"

3.3.1.3 2 seconds delay – PREF1_R2

Prerequisite: Set the delay between the reading of two words to 2 seconds

5) Review the following message "Hello, this a test for a gesture SMS service." (you might even give a paper note with the text to the participant to make sure they do not forget what is the message)
6) Replace "gesture" with "talking"

3.3.2 Listing contacts

3.3.2.1 1 second delay – PREF1_C1

Prerequisite: Set the delay between the reading of two contacts to 1 second

1) List all the contact beginning with a "P", and choose "Patricia"

3.3.2.2 1.5 second delay – PREF1_C1.5

Prerequisite: Set the delay between the reading of two contacts to 1.5 second

1) List all the contact beginning with a "P", and choose "Patricia"

3.3.2.3 2 seconds delay – PREF1_C2

Prerequisite: Set the delay between the reading of two contacts to 2 second

1) List all the contact beginning with a "P", and choose "Patricia"

4 Scheduling of the tasks

4.1 PERF1

PERF1_1:

PERF1_H1 - PERF1_N1 - PERF1_W1- PERF1_H2 - PERF1_N2 - PERF1_W2 - PERF1_H3 - PERF1_N3 - PERF1_W3

PERF1_2:

PERF1_H2 - PERF1_N2 - PERF1_W2 - PERF1_H3 - PERF1_N3 - PERF1_W3 - PERF1_H1 - PERF1_N1 - PERF1_W1

PERF1_3:

PERF1_H3 - PERF1_N3 - PERF1_W3 - PERF1_H1 - PERF1_N1 - PERF1_W1 - PERF1_H2 - PERF1_N2 - PERF1_W2

4.2 PERF2

PERF2_1:

PERF2_H1 – PERF2_N1 – PERF2_W1- PERF2_H2 – PERF2_N2 – PERF2_W2 – PERF2_H3 – PERF2_N3 – PERF2_W3

PERF2_2:

PERF2_H2 – PERF2_N2 – PERF2_W2 – PERF2_H3 – PERF2_N3 – PERF2_W3 – PERF2_H1 – PERF2_N1 – PERF2_W1

PERF2_3:

PERF2_H3 – PERF2_N3 – PERF2_W3 – PERF2_H1 – PERF2_N1 – PERF2_W1 – PERF2_H2 – PERF2_N2 –

Appendix B

Designing Accessible Mobile Interaction - A Case Study of a Text Messaging Application

This paper is mainly based on the prototype detailed in chapter 5. It was written in collaboration with David Costa, a LaSIGE's PhD student, Pr. Carlos Duarte and Pr. Bruno Dumas.

The paper was submitted to the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2016).

Designing Accessible Mobile Interaction – A Case Study of a Text Messaging Application

First Author
Address1
Address2
Address3
email@provider.com

Second Author
Address1
Address2
Address3
email@provider.com

Third Author
Address1
Address2
Address3
email@provider.com

Fourth Author
Address1
Address2
Address3
email@provider.com

ABSTRACT

While mobile devices have experienced important accessibility advances in the past years, people with visual impairments still face important barriers, especially in specific contexts when both their hands are not free to hold the mobile device, like when walking outside. By resorting to a multimodal combination of body based gestures and voice, we aim to achieve full hands and vision free interaction with mobile devices. In this paper we describe this vision and present the design of a prototype, inspired by that vision, of a text messaging application. The paper also presents a user study where the suitability of the proposed approach was assessed, and a performance comparison between our prototype and existing SMS applications was conducted. Study participants received positively the prototype, which also supported better performance in tasks that involved text editing.

CCS Concepts

•**Human-centered computing** → **Accessibility design and evaluation methods;** *Interaction techniques; Empirical studies in accessibility;*

Keywords

Non-visual interaction; body based interaction; gestures; voice; multimodal interaction; accessibility

1. INTRODUCTION

Today's mobile devices are flat touchscreens where you interact through touch and gestures. While this seems simple and easy for the average sighted person, persons with visual impairments have increased difficulties to do standard operations with their mobile phones. However, great accessibility solutions have been developed during the last decade. Current solutions consist in screen exploration techniques for navigation and text input. While they enable easier interaction for the blind, those techniques can still turn out to be very cumbersome and slow.

Novel technologies and research projects in input sensing have opened up new ways to interact with computers and other devices. Body interaction explores taps on skin, midair gestures or natural movements in order to trigger actions on an interactive devices. Moreover, performing actions that involve our own body does not require any form of visual attention. These techniques, combined with our proprioceptive capabilities, are interesting alternatives for interacting with, for example, mobile devices.

Still, the range of operations that are done today in mobile devices, introduce requirements that will not be dealt with body based interaction techniques. For example, text input is a common operation in mobile devices. For text input, other alternatives are, probably, more efficient than body based ones. To be able to address the multiple requirements, a multimodal solution combining different input modalities is more efficient.

This work addresses accessibility issues affecting persons with visual impairments when interacting with their smartphones. We introduce a new multimodal approach, based on on-body interaction and speech, aiming to improve the accessibility of smartphone's user interfaces. To demonstrate its applicability, we present the design of a prototype of a text messaging application following the proposed approach. We conducted a user study with 9 participants with visual impairments, where our prototype was compared with the current way text messages are entered by the study's participants. Not only did we receive positive feedback from the study's participants, we found that with our prototype they were able to perform different operations with a text messaging application quicker, even tough they were not experience with the proposed approach.

In the following section we present previous research on mobile accessibility, with a particular focus on text entry, and on body based interaction. Section 3 presents our approach to the design of accessible interaction in mobile devices. The user study is presented and its findings discussed

WOODSTOCK '97 El Paso, Texas USA

DOI: 10.475/123_4

in the next section. Finally, section 5 concludes the paper.

2. RELATED WORK

In a 2014 study [31], Ye et al. stated that 85% of visually impaired interviewed, own a smartphone. This is an interesting fact since touchscreens offer more accessibility barriers than non-smartphones. However, not only smartphones offer features that are not available in other phones, but visually impaired people also want to be trendy, even if they have to face interaction difficulties. Hence the importance of offering accessibility features and alternatives that accompany novel technologies.

2.1 Mobile Accessibility

A lot of research has been done in the last years to ease the interaction of visually impaired users with their smartphones, achieving better results than traditional methods. For instance, Kane et al. [15] presented an alternative solution to the standard features offered by mobile device's developers. The authors implemented Slide Rule, an interaction technique to improve the accessibility of multitouch screens when used by visual impaired users based on four main gestures.

Guerreiro et al. developed NavTouch [11, 12], a technique based on directional strokes to navigate through the alphabet in a way that decreased the cognitive load of memorizing the position of where the characters are located on the screen.

Bonner et al. [2] presented a novel solution for touchscreen's accessibility issues when used by visually impaired people. No-look Notes is an eyes-free text entry system that uses multi-touch input and auditory feedback. It offers a two-step access to the 26 characters of the alphabet with a small number of simple gestures and an 8-segment pie UI that removes the precise targeting required, for example, by QWERTY keyboards layouts.

Southern et al. [28] proposed BrailleTouch, a six-key chord braille keyboard for touchscreens. This technique was designed to use the smartphone's screen faced away from the user and held by the two hands. The screen was divided in 6 parts in the same way that BrailleType did and other control keys such as space, backspace or enter, were implemented through flick gestures.

In addition to text input via Braille Cells proposed in BrailleTouch, Holibraille [22] offers multi-touch capabilities to perform text editing (e.g. navigation through words, text selection). Two finger gestures combined with non-dominant hand gestures allow the user to edit and navigate in the text. The same authors also developed a novel correction system for these types of input [21].

2.2 Body Interaction

Recent technological advances and research projects about wearable technology and sensing devices have opened up new possibilities for using our body as an interaction platform. The always present skin, when combined with our proprioceptive capabilities, plus the control we can exert over our limbs are a sound alternative for interacting with personal devices, such as smartphones. This is an area with a great potential for improving the accessibility of interactive systems for multiple population groups, which is demonstrated by some recent works on this topic.

Virtual Shelves [16] uses the proprioceptive capabilities to support eyes-free interaction by assigning spatial regions centered around the user's body to applications shortcuts. To measure the different angles (theta and phi planes, i.e., up-down movement and left-right movement) between the body and the arm holding the cellphone, the system uses an accelerometer and a gyroscope.

In [17], Li et al. present an extension of Virtual Shelves [16]. This interaction technique leverages proprioception to access application shortcuts. However, it was not meant for visually impaired people to use. Therefore the authors performed a study to measure the directional accuracy of visual impaired persons and adapt the system for them. Also, the original work was not intended for mobile devices while the scope of this work was to enable mobile interaction (e.g. launch applications) when walking on the street.

Oakley and O'Modhrain [23] developed a motion based vibrotactile interface for mobile devices. The authors use 3-axial acceleration sensing to directly control list positions, instead of using this sensor to control the rate of scrolling or directional movement. The goal is to link or associate certain specific orientations to specific list items.

Dementyev and Paradiso [7] developed WristFlex, a system that makes use of pressure sensors to detect pulse gestures and is capable of distinguishing subtle finger pinch gestures with accuracy over 80%.

In [20], Matthies et al. describe a novel interaction technique, Botential, that makes use of sEMG and capacitive sensing, placed on a wristband, to detect different the body parts that it is in contact with.

Other recent approaches, such as Skinput [13] and PUB [18] use the skin as a means of interaction. These projects use bio-acoustics and ultrasonic signals respectively, to locate finger taps on body.

Latest efforts by Makino et al. [19] go beyond just touching and tapping actions, by using a new technique with photo reflective sensors, enabling pinching, pulling, twisting and caressing.

Dezfuli et al. [8] present a palm-based imaginary interface to control the TV. While the average effectiveness of the palm touches was around 96.8%, to track the hand movements and gestures a Kinect was used, making this not suitable for mobile environments.

2.3 Social Acceptability

These novel interaction techniques were also subject of several social acceptability studies. The willingness to perform these gestures will largely be dictated by how appropriate those actions look and feel when performed in public. Profita et al. [24] and Rico et al. [26, 25] findings show that there is a significant relation between audience and location with the willingness to perform certain gestures and this factor must be taken into account when building such systems. Findings also show that users are more fond of subtle movements, movements that are similar to what already exists in current technology (e.g. gestures similar to the ones in touch devices), and movements similar to the ones used in our everyday lives and enjoyable movements. On the other hand, participants in these studies stated that uncommon, large or noticeable movements would look weird in public settings.

In a work previous to these social acceptability studies, Constanza et al. [6] already used surface electromyography (sEMG) signals to capture subtle gestures, ones that are difficult for observers to perceive when someone is performing them.

Williamson et al. [30] developed a multimodal application with gesture recognition and audio feedback (speech and sounds). Despite being based on subtle gestures, the authors observed that some gestures were considered unacceptable in certain settings by the participants in the study. Consequently, the study participants developed new ways of performing the same gesture. To address this issue, such systems much be flexible and develop correction mechanisms in the recognition process. Additionally, the authors found that the willingness to perform gestures in a public setting do not depend only on the type of audience, but also if they are a sustained spectator (e.g. other passenger on a bus) or a transitory spectator (e.g. a person walking in the street).

3. DESIGNING MOBILE ACCESSIBLE INTERACTION

Although there is an increasing number of projects emerging in the body interaction field, its potential for improving mobile accessibility is being underestimated so far. We propose to explore body based interaction techniques, combined with speech in a multimodal system [9], towards this goal.

When exploring solutions designed for the visually impaired population, one must make sure that interaction is reliable, robust and adaptive to the user and the different contexts of use. In the proposed design space, it is necessary to understand how the human body can be used as a mean of interaction considering in particular the characteristics of this user group. Albeit vision is considered the primary spatial reference, there is no consensus [14, 29] if the level of spatial awareness within the body range is affected negatively or positively by vision impairments. The ability of accurately tapping in very specific parts of the body (e.g. dominant index finger tapping on the distal phalanx of the ring finger in the other hand) can also be affected by how long the person is blind or by the training received to develop proprioceptive skills. Moreover, as people with visual impairments do not have full notion of their surroundings, some gestures may be considered uncomfortable to perform in public settings (e.g. while pointing may hit another person).

Thus, the overall design process will require several steps. First, we have to study how body based techniques can improve the different activities performed in mobile devices. For that, it is necessary to understand their limitations. UI (user interface) navigation and text input on mobile devices are examples of activities we will focus on. We will investigate if a body-based interaction technique is appropriate for single or multiple activities. For instance, is a technique suitable only for application navigation (e.g. sliding a finger over the forearm to move through interactive elements) and target selection (e.g. performing a midair gesture to select an item), or can it be also used for content input.

Second, we will explore where the novel interaction techniques can replace existing techniques, where they cannot, and where they can complement those. Thus it is important to understand what existing traditional input modalities can be replaced or complemented by this novel interaction technique, in a manner that supports using the the same degree of function with those devices.

Finally, we want to consider how body interaction can be used as an intelligent interaction mechanism. Given that perceptual, motor and cognitive capabilities vary from person to person, and in the case of visually impaired persons many factors can affect those capabilities (e.g. born blind, disease that affected vision, etc.), we want to study mechanisms of adaptation to suit the complexity of the mapped interactive points on skin and defined gestures to the user capabilities.

To explore these interaction techniques, a set of prototypes will be iteratively designed, in collaboration with end user representatives, i.e., it will be developed based on a user centered design. Depending on the requirements, these prototypes can target interaction or technological validation. In the latter group, prototype can range from acoustic sensors capable of capturing and defining a large set of positions on the hand or forearm; light sensitive sensors that capture skin displacement when sliding for instance the finger on the forearm; to sensors sensitive to muscle movements which can capture gestures or movements. For the former group, interactive techniques will be trialed in multiple fidelity prototypes, sometimes resorting to Wizard of Oz techniques, whenever the technology is not at an adequate readiness level.

Our hypothesis is that by combining body based gestural and voice input we are able to increase the performance of users with visual impairments when interacting with mobile devices. One possible implementation is assigning gestures to specific application commands or shortcuts to certain applications. Through this we aim to reduce the time a user spends navigating the smartphone UI until he reaches the desired application or contextual command. UI navigation can represent a significant percentage of the interaction time for the visually impaired, thus our focus on a feature with a large potential to be useful. While it is true that the proposed solution may increase the users' cognitive load, we intend to take advantage of being able to memorize a set of gestures without overwhelming them. With the current accessibility solutions, users also end up memorizing the number of navigation steps required to reach the desired option to speed the navigation process.

In what concerns text input, we propose to use voice recognition in substitution of the current traditional methods (e.g. QWERTY keyboard display). The goal is to rely on the increasingly robust and accurate voice recognition technology to decrease the time and effort that visually impaired people have to put into writing a text. Nevertheless, in the future we will consider body based solutions for text input, like those based on, for instance, the Lorm alphabet [10, 4]. By offering multiple modalities we will address the need for text input in contexts where speech recognition could not be socially or technically feasible.

Combining these two forms of input, body-based and voice, we are able to offer an eyes and hands-free input interaction with the mobile device, making it suitable to interact in multiple contexts. In the next section we describe the design of a prototype of an application for text messaging that takes advantage of these combined input modalities.

3.1 Case Study: SMS application

Sending text messages is one of the most basic and most used features in every mobile device. This section presents a first prototype of an SMS application based on non traditional interaction methods, body-based gestures and voice recognition.

Inspired by the interaction approach presented above, a multimodal SMS application for the Android platform was developed, offering the standard SMS service features:

1. The user can enter a message;
2. The user has the possibility to review and edit the message if necessary;
3. The user can find and select the message addressee.
4. The user can send the entered message to the selected addressee.

Following our approach, by not relying on a visual interface, the user should be able to access every functionality without the need to navigate through a list of commands. To meet this goal, different actions are available to the user, with each action triggered by a specific gesture. This approach has the added benefit of allowing users to command the mobile application without having to hold the smartphone on their hands.

Current solutions of gesture recognizers based on sEMG sensors, like the one presented in [5], could prove useful for this scenario. However, they still are not capable to reliably address a per-user on-the-fly customizable set of gestures. Therefore, for our prototype we decided to use the Wizard-of-oz technique in the user studies. Gestures made by study's participants are interpreted by a human that inputs the command in the application. Voice commands are recognized using the speech recognizer available in the Android API.

The possible actions are triggered by gestures, with commands being a combination of gestures and voice input:

1. Add a message: After recognizing the "add a message" gesture, the speech recognizer is turned on and the user is notified by a sound. The user then dictates the message.
2. Review the message: After recognizing the "review message" gesture, the text to speech engine is turned on and the message is read word by word.
3. Edit a specific word in the message: During the message review, the user can stop the speech engine (by performing a gesture) whenever she wants to edit a word. Whenever that gesture is recognized, the application speaks the chosen word and turns on the speech recognizer. The user then dictates the new word. After that, the system informs the user that she will have to choose between five proposals. The application reads the proposals and the user stops it (by making a gesture) whenever she wants to select the desired proposal. The application tells the selected proposal and the change is carried out in the message.
4. Enter a token to search for a contact: After recognizing the "enter contact" gesture, the speech recognizer is turned on and the user says the name of the contact or a word that begins with the same letter of the contact's name. Then the application speaks how many contacts match this search. The user can add other tokens to refine the search, or proceed to hear the names returned by the search.
5. List the contact that match the current search: The application reads all the contacts that match the search token previously entered and the user can stop the application to select the desired contact. The application orally notifies the user of the chosen contact.
6. Send message: After recognizing the gesture the message is sent and the user is notified.

The design decision of presenting five proposals during the edition is to counter possible miss-recognition of the speech recognizer. The same motivation was behind the decision to consider only the first letter when entering a search token. Special attention was given to the message editing process to make it more accessible. Indeed, a survey of Azenkot et al. [1] showed that people with visual impairment spend 80% of their time editing when using speech recognition which can be frustrating.

4. USER STUDY

The conducted user study had two main goals. The first was to understand and study the behavior of visually impaired persons when interacting with the proposed approach, i.e., observe what type of gestures would persons with visual impairments perform for certain tasks and if they would feel comfortable to do so in different contexts. The second goal of the study was to compare, performance wise, the developed prototype with the applications that participants currently use to send text messages.

4.1 Methodology

To address the study's goals, we collected both quantitative and qualitative data through questionnaires, interviews and task observation.

The first step was to complete a short characterization questionnaire divided in two sections: Personal information and Mobile phone usage habits. A final question was asked in order to perceive the acceptability of doing midair gestures in a public setting.

In the next step, the performance of the prototype and the standard text message application used by the participants were compared. Participants were randomly split into two groups. One group begun by using their standard messaging application. The other group begun by using the prototype.

Before using the prototype, all participants were asked to define gestures for the available commands: 1) Start Application; 2) Write message; 3) Review message; 4) Edit specific word during the reading; 5) Confirm a proposal; 6) Enter a search token to find the contacts; 7) List the contents that match the search token; 8) Confirm one of the contacts while reviewing the list; and 9) Send the message. Participants were told they had no constraints about the gestures they could select, i.e., they could do mid-air gestures, gestures on their body, or whatever they found appropriate. All participants defined and performed all tasks while seated on a chair. Being a novel form of interaction, before completing the tasks in the prototype participants were given a training task to get acquainted with the flow of the application and recall all the gestures they defined moments before.

Participants were then asked to perform the same set of tasks in both the prototype and their usual messaging application. To compare their performance, a quantitative evaluation was designed based on time to perform different tasks on both applications. We provided a smartphone with our prototype for one of the conditions. For the other condition, participants used their own smarthpones and their preferred messaging application. We requested access to the participants' smartphone before starting the tasks to collect data about their list of contacts in order to make sure that all tasks had similar difficulty. Three different tasks with increasing difficulty levels were defined:

1. *Easy* task: The participant is asked to enter a text message (the message was provided). The participant is requested to review the message entered. No edition of the message is requested. The addressee is easily findable (maximum 3 results in the prototype condition and a contact figuring in the beginning or the end of the contact list for the standard application);

2. *Normal* task: The participant is asked to enter a text message (the message was provided). The participant is requested to review the message entered. The edition of 1 word of the message is asked. The addressee is moderately easy to find (maximum 5 results in the prototype condition and a contact figuring in the first or last quarter of the contact list for the standard application);

3. *Hard* task: The participant is asked to enter a text message (the message was provided). The participant is requested to review the message entered. The edition of 2 words of the message is asked. The addressee is more difficult to find (more than 5 results in the prototype condition and a contact figuring in the middle the contact list with a name beginning with a frequent letter for the standard application).

The first application used in the trial and the order of the three tasks were randomized for each participant. Each task was timed by an observer, registering the time each task and sub task (e.g. write message, review, etc.) took to complete and the errors made.

An evaluation concerning the delay between the reading of two words during message review or contact listing was also performed. Participants were asked to choose between 3 possible delays (1s, 1.5s and 2s) the one preferred. The order of presentation of the delays was randomized.

The final step consisted in filling two SUS (System Usability Scale) [3] based satisfaction questionnaires: one for the standard application and one for the prototype.

Given the duration to complete all tasks, approximately 1 hour and 45 minutes, it was decided to split the tasks by application into two sessions of one hour each in two different days.

4.2 Results

In this section we present the results found in the user study.

4.2.1 Participants

The group of participants was composed by 9 persons (3 females). Their ages ranged from 24 to 56 years old ($M = 32.78, SD = 9.62$). Eight of the participants lost their vision between 14 and 31 years old and one has partial vision since birth.

Regarding mobile phone consumption habits, 6 participants stated to own a smartphone for less than 2 years. Five persons own an Android phone and four an iPhone. When asked in what level they consider to be using the assistive technology built in (Talkback or VoiceOver), 3 considered themselves experts, 3 proficient, 2 average and one a beginner. In the specific case of using the messaging application, 2 considered to be experts while 4 of them considered to be proficient, 2 average users and 1 considered himself as a beginner. Nearly all participants send and receive messages everyday (8 out of 9).

When asked if when walking outside, they stop to answer an incoming call or an SMS or they ignore it, 6 participants stated to answer while walking. Regarding the acceptability of performing midair gestures in any context, public or not, 4 participants stated they would do it. The other 5 participants stated that it would depend on the gestures and location.

4.2.2 Gestures definition

Participants were asked to define a gesture for each one of the 9 used commands. In total, 81 gestures were registered. Despite the high number of gestures, few were repeated among participants. Nevertheless, we could observe a trend for 4 commands:

- **Adding text**. Movement similar to writing with the fingers on leg (figure 1, left);
- **Select a proposal**. Index finger touch (figure 1, center);
- **Selecting a contact during the list review**. Index finger touch (figure 1, center);
- **Send message**. Swipe Right (figure 1, right).

Other interesting findings were observed in the gesture definition phase. For instance, all participants concluded that using the same gesture for confirmation and selection purposes would be the best solution, decreasing the number of gestures to memorize. Regarding the gesture's visibility, 3 out of 9 participants defined subtle gestures, with most made beside the participant's pockets or on an available surface.

Additionally, we observed that several gestures meant different actions depending of the participant:

- **Thumbs up**. One participant considered this the gesture to start writing the message while other participant chose this gesture to send the SMS.
- **Swipe up with two fingers**. This gesture was defined to review the message but also to listen to the contacts list.
- **Drawing the letter C**. One participant chose this gesture to enter a search token to find a contact while other participant did the same gesture to listen to the contacts list that matched with the token.
- **Index finger double touch**. This gesture was defined to start writing the message and also to listen to the list of contacts that matched the search token.
- **Point with index finger**. One participant chose this gesture to open the application while other chose it to start writing the message.

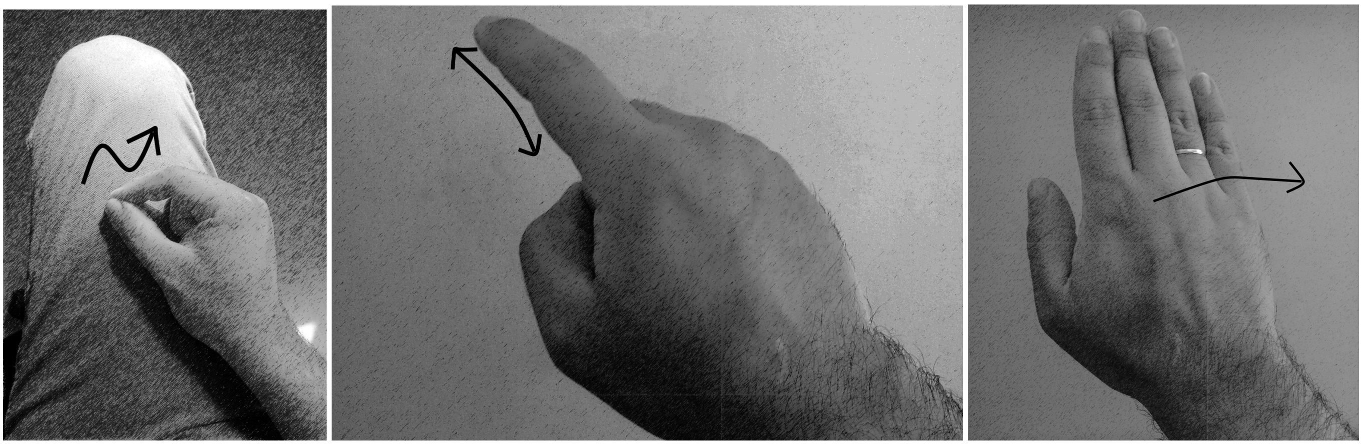

Figure 1: Most common gestures for writing a message (left), confirming a selection (center) and sending a message (right)

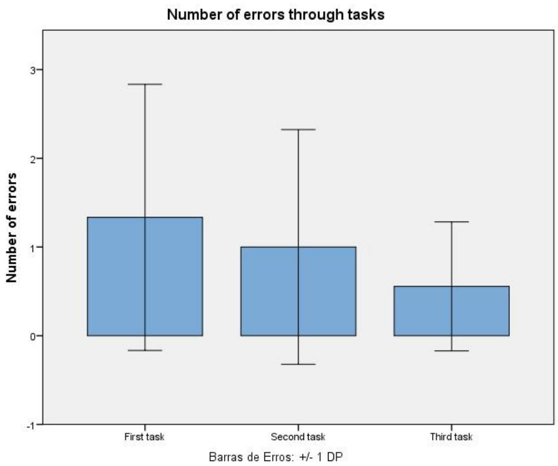

Figure 2: Average number of errors by task

4.2.3 *Learning effect*

While performing the tasks on the prototype, 26 gesture related errors were registered. We considered errors: 1) forgetting what gesture to do to trigger an action; 2) performing the wrong gesture; 3) selecting a wrong list item due to taking too long to recall the gesture. On average 0.96 errors occurred per participant ($SD = 1.20$).

It is interesting to analyze if those errors diminished over time. We registered the number of errors during the first task ($M = 1.33, SD = 1.5$), second task ($M = 1.00, SD = 1.32$) and third task ($M = 0.56, SD = 0.73$) as depicted in figure 2. Observing the mean values, data suggests that participants progressively memorized the gestures needed to complete the tasks. For this reason, a statistical analysis was conducted. In order to check for any significant differences between task errors, we began by assessing the normality of the data. The results for all tasks showed that normality was not verified ($p_1 = 0.077; p_2 = 0.005; p_3 = 0.008$). Consequently, we conducted a non-parametric Friedman test, comparing the errors made in each task, which revealed that there was not a statistically significant difference ($\chi^2(2) = 1.826; p = 0.401$).

4.2.4 *Reading speed preferences for selection tasks*

Participants were asked to rate their preference of 3 different reading speeds on two contexts: selecting a word from the message; and selecting a contact from the list of contacts. The purpose was to understand how fast they are able to react and perform the gesture to select a word or contact.

Findings show that despite 22% of the participants fail to select the right word while using the fastest speed, over 55% preferred this option. Those who failed stated they could do it with more training. On the other hand, when selecting a contact from the list, there was not a preferred reading speed choice. Four participants preferred the fastest speed and other 4 preferred the middle one. However, 33% of the participants failed when selecting a contact in the fastest speed mode.

4.2.5 *Task execution times*

Figure 3 presents the time taken to complete the three tasks with both applications. When using the developed prototype, participants took, on average, 34.78 ($SD = 10.77$) seconds to complete the easy task. For the normal task, participants took, on average, 83.22 ($SD = 20.94$) seconds to finish it. Finally, for the hard task, participants needed 118.89 ($SD = 21.03$) seconds, on average, to complete it. When using the standard SMS application, participants took 39 ($SD = 21.54$) seconds to complete the easy task, on average. Participants completed the normal task in 146.11 ($SD = 66.08$) seconds, on average. The hard task took, on average, 238.11 ($SD = 112.74$) seconds to finish.

By observing the data, two trends seem to emerge. More difficult tasks took more time to complete, as expected, and tasks were completed quicker when made with the prototype. To validate these observations we need to look for the effects of the independent variables (task difficulty and application) on the dependent variable (time to complete the task). We began by assessing the normality of the data. The results showed that the data follows a normal distribution except for one group (easy task on standard application).

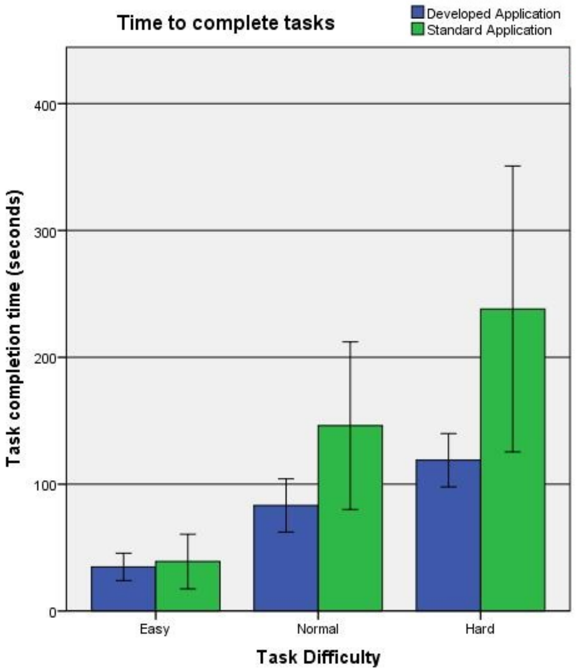

Figure 3: Time to complete tasks by difficulty level and application.

Because of its robustness regarding violation of normality, a parametric two-way repeated measures ANOVA test was still performed. Results show that there is a significant interaction between the two independent variables ($F(2, 16) = 8.024, p = 0.004$). In order to understand this interaction we need to determine the difference between the groups at each level of each factor.

The first factor to be analyzed was the application used. We computed a simple main effect test to observe the differences between the two applications for each level of task difficulty. Results showed that for the easy task we could not find any significant difference between the two applications ($p = 0.664$). However, we found statistical differences in the normal ($p = 0.026$) and hard ($p = 0.012$) tasks. We can conclude that our prototype allowed the study participants to complete the tasks quicker than the standard SMS application in the tasks that required message editing, i.e., the normal and hard tasks. In the task that simply required text input and the selection of a contact in the beginning of the list of contacts there were no differences between the applications, even tough, on average, our prototype was still more than 4 seconds quicker than the standard application.

The second factor analyzed was the task difficulty. We ran another simple main effect test to assess the differences between task difficulty within each application. Observing the pairwise comparisons, the results show a significant difference for easy and normal tasks ($p < 0.001$), easy and hard tasks ($p < 0.001$) and normal and hard tasks ($p = 0.001$) for the prototype. The same results occurred in the standard application. Findings show that there is also a significant difference between easy and normal ($p = 0.001$) tasks, easy and hard tasks ($p < 0.001$) and normal and hard tasks ($p = 0.012$). Regarding task difficulty, we can conclude that more difficult tasks take more time to complete, independently of the application used, which is an expected result.

4.2.6 *SUS analysis*

Both applications rated high on the SUS scale. The prototype averaged 74.44 points, while the standard SMS applications rated, on average, 74.72 points. To check for any statistically significant differences regarding the SUS scores for both the standard applications and the developed prototype, we first assessed the normality of the data. The results of Shapiro-Wilk test showed data with a normal distribution for both conditions. A paired t-test showed that there was not a significant effect ($t(8) = -0.0322, p = 0.975$) on the SUS score when comparing the results between the standard SMS applications ($M = 74.72, SD = 20.78$) and the prototype ($M = 74.44, SD = 10.66$).

4.3 Discussion

One of our goals was to understand how participants would react to a different form of interaction with their smartphones, namely one that involved performing gestures but not on their smartphones' touchscreen. Our findings are in concordance with previous work by Profita et al. [24] and Rico et al. [26, 25]. Our subjects are conscious of their actions in a public setting and some state they would feel embarrassed performing such gestures. However, if those gestures could be performed more subtly they are willing to use them if that would bring interaction advantages.

Regarding the definition of gestures, the most significant conclusion is that every confirmation command should be triggered by the same gesture, independently of the context. This supports a consistent interaction and reduces the number of gesture to memorize. Another significant observation is that We could hardly find repeated gestures for the same action across participants. Even though the sample of participants recruited is small, the diversity of proposed gestures leads to the conclusion that, for this kind of interaction, users need to be able to define their own gestures, thus the interface should support customization. Still, one participant suggested to replicate the already known touch gestures, for the actions where this solution could be applied. This would reduce the cognitive load and the need to learn new gestures.

The majority of participants preferred quicker reading velocities despite not being able to select an item in time, arguing that with enough training they would be able to perform correctly. This opens up an opportunity to assist users, through a correction algorithm that would adapt the selection as a function of the users' reaction time.

When comparing participants' performance in their usual text messaging application and a prototype developed following our proposed approach, we were able to find advantages in our prototype. While there were no significant differences when the task required no editing of the message, we found that our prototype supported quicker message correction than the standard solutions available. This results from the careful design of the message navigation and correction mechanisms. This prototype was designed to take advantage of the possibility to have commands issued by the user, through on-body gestures, while aurally inspecting the entered message. Additionally, since those commands need not be entered via the smartphone's touchscreen, users are able to instantly input different commands, instead of having to navigate a list of commands, which is the current paradigm.

Regarding the brief usability assessment, through the SUS questionnaire, findings showed no statistically significant difference between the standard SMS application and the developed prototype. Still, both achieved a high SUS score of 74 points. According to the SUS scoring system [27], results higher than 68 are above average therefore considered usable. Additionally, 5 participants stated that *"this application is much easier than the one that already exist for the blind"* and two other participants said it to be *"very functional upon learning the gestures."*. Moreover, 8 participants stated that the *"keyboard is too small which can lead to errors while writing the message"* when discussing current applications. However, two participants raised concerns about using an online voice recognizer, stating the limitations of being required to be always connected to the internet because of the lack of performance of current offline versions.

5. CONCLUSION

Despite current great accessibility options offered in both Android and iOS mobile systems, they still lack decent support for some types of activities and contexts of usage. For instance, writing text is still a time-consuming task for users who suffer from some kind of visual impairment. Besides, to write a message, users currently have to use both hands. This is particularly important when considering this user group since they need to hold a cane or a guide dog in one hand while walking.

Our approach envisions to make the interaction with mobile devices 100% hands free by exploring the advantages of multimodal techniques, specially the use of mid-air and on-body gestures combined with voice recognition. While the ultimate challenge is to design an interaction model for the whole system, this paper presented an initial case study focused on an application for the sending of text messages. A user study was conducted and results showed positive feedback from the participants regarding the use of body based interaction. We also found the proposed design to have a better performance when text editing was required, when compared with current available text messaging applications. Other findings raised the need for socially acceptable or conspicuous gestures, and the usefulness of customized and adaptive interaction techniques.

Reserved for the future is the integration of a fully functional gesture recognizer which takes into account the feedback obtained during this first experiment (e.g. subtle gestures, possibility to customize gestures). Moreover, we intend to expand and generalize this novel interaction method into the whole operating system, making it available as an accessible service so that all applications can benefit from it.

6. REFERENCES

[1] Shiri Azenkot and Nicole B. Lee. Exploring the use of speech input by blind people on mobile devices. *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '13*, pages 1–8, 2013.

[2] Matthew N. Bonner, Jeremy T. Brudvik, Gregory D. Abowd, and W. Keith Edwards. No-look notes: Accessible eyes-free multi-touch text entry. In *Proceedings of the 8th International Conference on Pervasive Computing*, Pervasive'10, pages 409–426, Berlin, Heidelberg, 2010. Springer-Verlag.

[3] John Brooke. *Usability Evaluation in Industry*, chapter SUS: a quick and dirty usability scale, pages 189–194. Taylor and Francis, London, 1996.

[4] Nicholas Caporusso, Michelantonio Trizio, and Giovanni Perrone. *Pervasive Health: State-of-the-art and Beyond*, chapter Pervasive Assistive Technology for the Deaf-Blind Need, Emergency and Assistance Through the Sense of Touch, pages 289–316. Springer London, London, 2014.

[5] David Costa and Carlos Duarte. From one to many users and contexts: A classifier for hand and arm gestures. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*, IUI '15, pages 115–120, New York, NY, USA, 2015. ACM.

[6] Enrico Costanza, Samuel A. Inverso, Rebecca Allen, and Pattie Maes. Intimate interfaces in action: Assessing the usability and subtlety of emg-based motionless gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 819–828, New York, NY, USA, 2007. ACM.

[7] Artem Dementyev and Joseph A. Paradiso. Wristflex: Low-power gesture input with wrist-worn pressure sensors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 161–166, New York, NY, USA, 2014. ACM.

[8] Niloofar Dezfuli, Mohammadreza Khalilbeigi, Jochen Huber, Florian Müller, and Max Mühlhäuser. Palmrc: Imaginary palm-based remote control for eyes-free television interaction. In *Proceedings of the 10th European Conference on Interactive Tv and Video*, EuroiTV '12, pages 27–34, New York, NY, USA, 2012. ACM.

[9] Bruno Dumas, María Solórzano, and Beat Signer. Design guidelines for adaptive multimodal mobile input solutions. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '13, pages 285–294, New York, NY, USA, 2013. ACM.

[10] Ulrike Gollner, Tom Bieling, and Gesche Joost. Mobile lorm glove: Introducing a communication device for deaf-blind people. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction*, TEI '12, pages 127–130, New York, NY, USA, 2012. ACM.

[11] T. Guerreiro, P. Lagoa, H. Nicolau, D. Gonçalves, and J.A. Jorge. From tapping to touching: Making touch screens accessible to blind users. *MultiMedia, IEEE*, 15(4):48–50, Oct 2008.

[12] Tiago Guerreiro, Paulo Lagoá, Hugo Nicolau, Pedro Santana, and Joaquim Jorge. Mobile text-entry models for people with disabilities. In *Proceedings of the 15th European Conference on Cognitive Ergonomics: The Ergonomics of Cool Interaction*, ECCE '08, pages 39:1–39:4, New York, NY, USA, 2008. ACM.

[13] Chris Harrison, Desney Tan, and Dan Morris.

Skinput: Appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 453–462, New York, NY, USA, 2010. ACM.

[14] B. Jones. Spatial perception in the blind. *British Journal of Psychology*, 66(4):461–472, 1975.

[15] Shaun K. Kane, Jeffrey P. Bigham, and Jacob O. Wobbrock. Slide rule: Making mobile touch screens accessible to blind people using multi-touch interaction techniques. In *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility*, Assets '08, pages 73–80, New York, NY, USA, 2008. ACM.

[16] Frank Chun Yat Li, David Dearman, and Khai N. Truong. Virtual shelves: Interactions with orientation aware devices. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*, UIST '09, pages 125–128, New York, NY, USA, 2009. ACM.

[17] Frank Chun Yat Li, David Dearman, and Khai N. Truong. Leveraging proprioception to make mobile phones more accessible to users with visual impairments. In *Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '10, pages 187–194, New York, NY, USA, 2010. ACM.

[18] Shu-Yang Lin, Chao-Huai Su, Kai-Yin Cheng, Rong-Hao Liang, Tzu-Hao Kuo, and Bing-Yu Chen. Pub - point upon body: Exploring eyes-free interaction and methods on an arm. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 481–488, New York, NY, USA, 2011. ACM.

[19] Yasutoshi Makino, Yuta Sugiura, Masa Ogata, and Masahiko Inami. Tangential force sensing system on forearm. In *Proceedings of the 4th Augmented Human International Conference*, AH '13, pages 29–34, New York, NY, USA, 2013. ACM.

[20] Denys J. C. Matthies, Simon T. Perrault, Bodo Urban, and Shengdong Zhao. Botential: Localizing on-body gestures by measuring electrical signatures on the human skin. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '15, pages 207–216, New York, NY, USA, 2015. ACM.

[21] Hugo Nicolau, Kyle Montague, Tiago Guerreiro, João Guerreiro, and Vicki L. Hanson. B#: Chord-based correction for multitouch braille input. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, pages 1705–1708, New York, NY, USA, 2014. ACM.

[22] Hugo Nicolau, Kyle Montague, Tiago Guerreiro, André Rodrigues, and Vicki L. Hanson. Holibraille: Multipoint vibrotactile feedback on mobile devices. In *Proceedings of the 12th Web for All Conference*, W4A '15, pages 30:1–30:4, New York, NY, USA, 2015. ACM.

[23] I. Oakley and S. O'Modhrain. Tilt to scroll: evaluating a motion based vibrotactile mobile interface. In *Eurohaptics Conference, 2005 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2005. World Haptics 2005. First Joint*, pages 40–49, March 2005.

[24] Halley P. Profita, James Clawson, Scott Gilliland, Clint Zeagler, Thad Starner, Jim Budd, and Ellen Yi-Luen Do. Don't mind me touching my wrist: A case study of interacting with on-body technology in public. In *Proceedings of the 2013 International Symposium on Wearable Computers*, ISWC '13, pages 89–96, New York, NY, USA, 2013. ACM.

[25] Julie Rico and Stephen Brewster. Gestures all around us: User differences in social acceptability perceptions of gesture based interfaces. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, pages 64:1–64:2, New York, NY, USA, 2009. ACM.

[26] Julie Rico and Stephen Brewster. Usable gestures for mobile interfaces: Evaluating social acceptability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 887–896, New York, NY, USA, 2010. ACM.

[27] Jeff Sauro. Measuring usability with the system usability scale (sus). http://www.measuringu.com/sus.php, 2011. Accessed: 06.05.2016.

[28] Caleb Southern, James Clawson, Brian Frey, Gregory Abowd, and Mario Romero. An evaluation of brailletouch: Mobile touchscreen text entry for the visually impaired. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '12, pages 317–326, New York, NY, USA, 2012. ACM.

[29] F. Thinus-Blanc and C. Gaunet. Representation of space in blind persons: Vision as a spatial sense? *Psychological Bulletin*, 121:20–42, 1997.

[30] Julie R. Wiliamson, Andrew Crossan, and Stephen Brewster. Multimodal mobile interactions: Usability studies in real world settings. In *Proceedings of the 13th International Conference on Multimodal Interfaces*, ICMI '11, pages 361–368, New York, NY, USA, 2011. ACM.

[31] Hanlu Ye, Meethu Malu, Uran Oh, and Leah Findlater. Current and future mobile and wearable device use by people with visual impairments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 3123–3132, New York, NY, USA, 2014. ACM.