Vibro-Tactile Enrichment Improves Blind User Interaction with Mobile Touchscreens

Maria Buzzi, Marina Buzzi, Barbara Leporini, Maria Paratore

Vibro-Tactile Enrichment Improves Blind User Interaction with Mobile Touchscreens

Maria Claudia Buzzi[1], Marina Buzzi[1], Barbara Leporini[2] and Maria Teresa Paratore[1]

[1] IIT - CNR, via Moruzzi 1, 56124 Pisa (PI), Italy
[2] ISTI - CNR, via Moruzzi 1, 56124 Pisa (PI), Italy

{Claudia.Buzzi, Marina.Buzzi}@iit.cnr.it,
Barbara.Leporini@isti.cnr.it, Maria.Paratore@iit.cnr.it

Abstract. Interaction via mobile devices is a challenge for blind users, who often encounter severe accessibility and usability problems. The main issues are due to the lack of hardware keys, making it difficult to quickly reach an area or activate functions, and to the new way of interacting via touchscreen. A touchscreen has no specific reference points detectable by feel, so a blind user cannot easily understand exactly where (s)he is positioned on the interface nor readily find a specific item/function. Alternative ways to provide content are mainly vocal and may be inadequate in some situations, e.g., noisy environments. In this study we investigate enriching the user interfaces of touchscreen mobile devices to facilitate blind users' orientation. We propose a possible solution for improving interaction based on the vibro-tactile channel. After introducing the idea behind our approach, two implemented Android Apps including the enriched user interfaces are shown and discussed.

Keywords: Accessibility, usability, mobile accessibility, haptic UIs, blind.

1 Introduction

In recent years, the development of more sophisticated smartphones with touchscreens has changed interaction modalities, while the use of hardware keys to quickly reach or activate specific functions has been decreasing. Touchscreens are completely smooth, so detecting specific user interface (UI) parts and elements is difficult for someone who cannot see the screen. Alternative ways to provide content visible on the touchscreen are mainly based on vocal channels through a voice TTS (Text-to-Speech), which may not always be enough for fully accessible and usable interaction. Vocal feedback may not work well in noisy environments; moreover, in some situations (e.g. during classes, meetings, speeches) users may prefer something more 'silent'.

Several aspects of interacting with a touchscreen can be improved to make screen exploration more satisfactory for the blind. In this paper we focus on a tactile-based solution to improve mobile interface exploration by blind users when interacting via touchscreen. The tactile channel is more immediate and direct for the blind -- cortical

adfa, p. 1, 2011.

brain areas normally reserved for vision may be activated by tactile stimuli [11]. In this perspective, a possible solution for improving interaction based on the vibro-tactile channel is presented. After an introduction to related works in the field and to the main usability issues encountered when a blind person interacts with a touchscreen, our approach will be discussed and described through examples.

2 Related Work

Several studies in the literature describe the importance of providing a user with appropriate mechanisms and techniques for better orientation on the user interface. In [4], the authors conducted a pilot study to investigate analogies between the real world and Web navigation. For the authors, organized content can benefit a reader only if (s)he is able to move around it with accuracy and agility, and is able to quickly discover and absorb its organization.

Kane et al. compared how blind and sighted people use touchscreen gestures, and proposed guidelines for designing accessible touchscreens. Blind subjects preferred gestures that use screen corners, edges, and multi-touch, enabling quicker and easier identification, and suggested new gestures in well-known spatial layouts, such as a qwerty keyboard [5]. This study observed that referencing points are particularly useful and preferred by blind users to better move around the interface. Arroba et al. proposed a methodology for making mobile touchscreen platforms accessible for visually impaired people based on a functional gesture specification, a set of guidelines to assure consistency of mobile platforms and the customization of input application [1]. These studies offered guidelines valuable for further developments.

Recently, Bonner at al. developed an accessible eyes-free text entry system that offers multi-touch input and audio output. The system, implemented on Apple's iPhone and tested with ten users, showed better performance in terms of speed, accuracy and user preference with respect to the text entry component of VoiceOver [2]. This solution focuses only on the text-entry providing audio feedback, which may not work well in some situations.

Koskinen et al. investigated the most pleasant tactile clicks, comparing piezo actuators vs a vibration motor, finding that subjectively the first was preferred [6]. In agreement with previous studies, results showed that tactile feedback improves the usability of virtual buttons pressed with the fingers, since the user is able to feel the object of interaction. Brewster and Brown proposed the use of a new type of tactile output: structured tactons, or tactile icons, i.e., abstract messages that can be used to communicate information. A tacton is characterized by parameters such as frequency, amplitude and duration of a tactile pulse, but also rhythm and location [3]. Using tactons could enhance accessibility of mobile devices for blind users as well as for sighted people in motion. Qian et al. identified the salient features of tactons when integrated with a mobile phone interface. Findings indicated that the best results use simple static rhythms, with differences in each pulse's duration. However, to ensure accurate perception, the dimensions in which paired tactons differ should be limited [10]. Yatani and Truong proposed the use of multiple vibration motors embedded in the back of the mobile touchscreen device to convey tactile feedback, providing semantic information about the touched object. They showed that users can accurately distinguish ten vibration patterns, and that the proposed system enables better interaction in an eyes-free setting than devices without tactile feedback or using only a single vibration motor [12]. However, these solutions mainly rely on hardware add-ons for providing haptic feedback while our approach offers a non-invasive/intrusive software solution, that has no impact on the usual interaction of blind users.

Other studies have investigated the use of tactile aids to enhance blind user interaction on touchscreen devices. In a preliminary study, Magnusson et al. investigated the use of haptic and speech feedback to make a digital map on a touchscreen more accessible [8], developing a prototype application that uses vibration to help blind users understand a map layout. This solution requires a time-consuming pre-processing of each map [9]. Our approach is conceptually similar to this last work since we use a mix of audio and vibration for easily detecting areas of interest on the user interface. However, despite all this research, to the best of the authors' knowledge, effective ad-hoc enrichment of general-purpose touchscreen user interfaces via software for easier orientation of blind persons has not yet been presented.

3 Interacting with a Touch Mobile Device: Usability Issues

To identify main accessibility and usability issues encountered by blind users when interacting with touchscreen mobile devices, we evaluated the interaction with Android-based and Apple mobile devices. In both cases, the inspection was carried out by all the authors (one of whom is blind) interacting with the touchscreen via screen reader (TalkBack on Android devices, VoiceOver on Apple ones) and gestures. Throughout the evaluation process, sighted authors avoided looking at the screen by activating the "screen curtain"[1] functionality. More details on the applied methodology are available in [7]. In the following we summarize the main issues observed:

- Lack of logical navigation order, to ensure the content is correctly sequentialized: the problem occurs when trying to navigate content and elements sequentially via swipe gestures ("next" - flick right - and "previous" - flick left). In this case some incongruences regarding the correct logical order occur when visiting or expanding an item.
- Unsuitable handling of focus: the problem especially occurs when editing a text field within a form composed of several control UI elements. When activating an edit field by a double tap, the system focus moves onto that field and the screen reader announces the editing modality. By exploring and clicking on the virtual keyboard letters, the focus moves on the keyboard and the user loses the editing focus point. (S)He is unable to quickly check what was edited because to do this, it is necessary to explore the UI again. This issue also arises when filling in a group of form elements. This process disorients the user and can make the action difficult and frustrating.

[1] http://www.apple.com/accessibility/iphone/vision.html

- Lack of orientation on the UI: the user can explore content on the screen by either (1) going forward and backward in a sequential order through swipe gestures, or (2) touching a point on the screen and then proceeding through the next and previous flick. This means that the user may encounter some difficulty or frustration when reading content. For instance, when reading a mail, the user has to read the message header before catching its content, unless (s)he is able to click in the exact starting point of the text of message. When clicking, the focus moves onto the clicked place and the user can start the reading from there.

4 The Proposed Solution

To address the usability issues discussed in the previous section, we propose an approach based on the use of haptic technology. Our proposal aims to support blind users as they explore and interact with content on a touchscreen. Preserving the original UI layout, this solution provides extra information and feedback for better and easy identification of the UI elements or parts.

For instance, for the Web, the W3C WAI-ARIA (Accessible Rich Internet Applications) suite[2] suggests the use of landmarks and region roles to allow developers to divide the Web page content into several parts, to create the logical partitioning of UI areas. In this way, the user is able to quickly obtain an overview of the page structure. Unfortunately this standard is still rarely applied. Moving from web to mobile devices, phone apps -- as well as any system application -- could greatly enhance applying a similar strategy to the main UI parts. In our study we decided to apply the “Logical partitioning” of UI elements to touchscreen interfaces of mobile devices to make the main parts of the UI easily and rapidly detectable. To this aim, we use reference cues (haptic mechanism) to help the user recognize those parts. Reference cues are particularly important for blind users in order to better orient themselves and move around a real and virtual space [4]. Depending on the UIs, haptic tags are added to help a blind user localize a specific part or elements on the interface.

To test our approach, we developed a prototype for an Android device, specifically a Samsung Galaxy Nexus running Android 4.2. The solution proposed herein leverages Android’s support for accessibility and aims to provide developers with a simple yet versatile tool that can be used to improve the UI usability for blind users. We will describe our approach in practice through some examples in Section 5.

4.1 Methodology

According to the Model-View-Controller (MVC) software design pattern, logic and presentation must be strongly separated. Any Android application should be 'natively' MVC-compliant, since Android development guidelines require UIs to be described entirely by means of XML files. We took advantage of this principle to develop a simple add-on that can be used to enrich our UIs. We defined a graphical object with

[2] http://www.w3.org/WAI/intro/aria

customizable behaviour in terms of accessibility. The goal is to offer developers a flexible tool for enriching UIs in order to improve their accessibility. XML attributes define the accessibility property of a cue and allow customizing any cue through spoken messages, sounds and vibrations. This improvement is nearly seamless since it will only require modifying an XML file to configure the cue's parameters. In the following, we refer to this kind of cue as CAC: Customizable Accessibility Cue. The XML snippet shown below represents a typical CAC, provided with a spoken message, a vibrating pattern and a sound. Once it is inserted into a layout XML file, it is rendered by Android as a ‘reactive’ colored ball, as shown in Fig. 1.

```
<org.cnr.iit.accessible.CustomAccessibleButton
  android:id="@+id/ballBtn01"
  android:layout_width="fill_parent"
  android:layout_height="20dp"
  android:layout_gravity="top"
  android:layout_margin="5dp"
  android:paddingTop="10dp"
  cnr:customMsg="custom spoken message"
  cnr.vibPattern="0,100,100,100"
  cnr:customSound="mySound" />
```

Fig. 1. A vibrating and speaking cue also associated with a sound, graphically rendered on a UI by a colored ball.

5 Applying the methodology to real cases

The proposed solution described above was tested on two real applications: a dial pad and an email client. The procedure was aimed at identifying the most suitable areas to put the cues and related feedback: a single event or a combination of events chosen from among a single vibration, vibrating pattern, speaking feedback and sound feedback, to better identify the different logical areas for each UI. We also improved the applications from other points of view, whenever their usability could be further improved. The main goal was to ‘mark’ the critical interaction areas by adding the customizable accessibility cues (CACs) to the layout XML files.

In the first prototype, the CACs to mark each logical session were points placed on the left side of the screen. A preliminary test with a blind user highlighted that this solution could have been misleading, since the hints were available only when exploring a narrow area of the screen. To resolve this issue, each CAC on the UI was ‘stretched’ (by means of proper dimension attributes) to become a horizontal strip. A pilot test was carried out with two blind users in order to assess and improve the proposed approach on two applications. The users were asked to perform five tasks using the Samsung Galaxy Nexus provided by the authors; the test was carried out in a laboratory applying the Think Aloud protocol. We first introduced users to our general idea of marking the main UI parts to support the exploration. No specific indication on how those markers were placed or implemented (audio and tactile) was given. The valuable feedback we received allowed us to refine both the haptic cue, size and vibration time, and the feedback provided, to enhance user interaction experience. All of the tests were performed after having enabled and suitably configured the Android's screen reader TalkBack.

5.1 The dial pad

In order to choose the better locations of the CACs, we identified the main interaction areas: the status bar (battery indicator, mobile network, time, etc.); numeric display area (where the number the user is dialing is visible), numeric keyboard (to dial the number), call and delete buttons; navigation bar (home and back buttons). Then we placed four CACs to separate these five areas. The uppermost and the lowest were associated with a sound both to highlight the boundaries of the part of the UI strictly related to the dial pad, and at the same time, marking the status -- at the top -- and the navigation -- at the bottom -- bars. These hints helped avoid accidentally pressing of one of the smartphone's built-in soft-keys that compose the status and the navigation bars. The intermediate cues were placed to highlight the other areas: the beginning and end of the numeric keyboard is marked by means of a vibrating pattern, thus making it easy to detect also the numeric display area and the call and delete buttons.

Fig. 2. The telephone app with the CACs introduced to improve its usability.

Blind users performed two tasks: (1) dialing a call and (2) checking the phone number on the numeric display area. Blind users were able to perform the tasks and reach the desired areas easily and rapidly. While performing our tests, we noticed that other enhancements could be made to improve the usability when dialing a number: the app was modified in order to have each number announced, both when hovering over the relative button and when double clicking it, i.e., when the number was selected. In order to check the correctness of the dialed numbers, a "spoken message event" was added when hovering over the number display.

5.2 The email client

K-9 is an open-source email client based on the Email application shipped with the initial release of Android. The application is quite complex and offers a complete environment in which to organize and compose emails. Once the appropriate XML layout files were identified, we introduced the cues according to the principle of highlighting main/critical interaction areas. The application was tested by two blind users considering three tasks: (1) browsing the list of incoming emails (2) reading an email and (3) composing an email. When browsing the inbox (1) it was necessary to border the UI to highlight the email list, with one CAC on the top and one on the bottom of the list. When reading an email (2) it was necessary to mark the email text; the cues were positioned as in (1). In the 'Compose' UI (3) a cue was placed before the soft-keyboard to separate it from the editing area. These cues were all associated with the same sound since they share the same function of 'UI border'.

Following the suggestions of the two blind users who tested the applications, other improvements were made to enhance the usability of text fields: a sound was associated with the event of a text field getting the focus, thus becoming suitable for filling. Moreover, to favor their detection, the text fields were announced by a 100-msec-vibration - normally -- or a 300-msec-vibration -- if the text field held the focus.

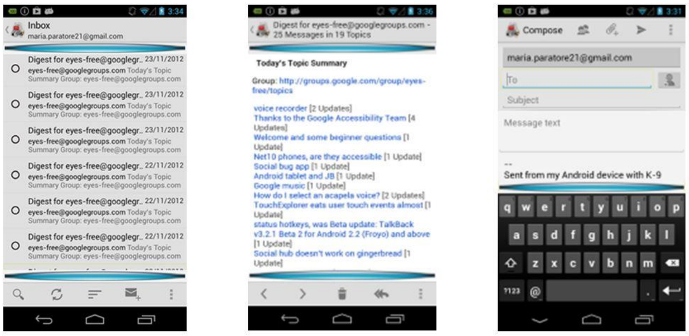

Fig. 3. Position of the CACs delimiting interaction areas in three UIs of K-9, respectively: mail browsing, single message view, mail composer.

6 Conclusions

In this work we discussed how the principle of the "Logical partitioning of UI elements" can be applied to a mobile interface in order to enhance touchscreen interaction for blind users. The proposed solution is based on a customizable combination of haptic and audio feedback that can be placed programmatically on UIs with usability problems. Only vocal feedback may not work well in noisy environments or cannot be used during classes, meetings, speeches, etc. The Customizable Accessibility Cue (CAC), a flexible add-on for enriching the UIs with spoken messages, sounds and vibrations, was developed. To test our approach, we considered two Android applications, a telephone and an email client. A pilot test was conducted with two blind persons, who provided enthusiastic and positive feedback concerning the usefulness of the proposed solution, and useful feedback for refining on the UIs.

Future work will include some improvement of the methodology, e.g., expanding the number of UIs and identifying the potential best set of CACs for each UI; using different short sounds to announce useful information, such as focus shifting from one area to another, or to provide an additional confirm for number editing on the phone keypad, etc. Furthermore, we need to make a user test with a group of blind persons in order to evaluate their performance interacting with both the simple and enriched user interfaces, gathering quantitative data for evaluating and improving the proposed solution, which is potentially applicable to touchscreens of any device.

7 References

1. Arroba, P., Vallejo, J.C., Araujo, A., Fraga D., Moya, J.M.: A Methodology for Developing Accessible Mobile Platforms over Leading Devices for Visually Impaired People. In: Ambient Assisted Living. LNCS 6693, pp. 209-215. Springer. DOI: 10.1007/978-3-642-21303-8_29 (2011)
2. Bonner, M., Brudvik, J., Abowd, G., Edwards, W.K.: No-Look Notes: Accessible eyes-free multi-touch text entry. In: Pervasive 2010, LNCS 6030, pp. 409-426. Springer (2010)
3. Brewster, S., Brown, L. M.: Tactons: structured tactile messages for non-visual information display. In: A. Cockburn (Ed.) Fifth Conf. on Australasian User Interface (AUIC '04), Vol. 28. Australian Computer Society Inc., Darlinghurst, Australia, pp. 15-23 (2004)
4. Goble, C., Harper, S., Stevens, R.: The travails of visually impaired web travellers. In: Hypertext 2000, pp. 1-10 (2000)
5. Kane, S.K., Wobbrock, J.O., Ladner, R.E.: Usable gestures for blind people: understanding preference and performance. In: CHI '11, pp. 413-422. ACM (2011)
6. Koskinen, E., Kaaresoja, T., Laitinen, P.: Feel-good touch: finding the most pleasant tactile feedback for a mobile touch screen button. In: ICMI '08, pp. 297-304. ACM, New York, NY, USA. http://doi.acm.org/10.1145/1452392.1452453 (2008)
7. Leporini, B., Buzzi, M. C., Buzzi, M.: Interacting with mobile devices via VoiceOver: usability and accessibility issues. In: OzCHI '12. pp. 339-348. ACM, New York, NY, USA, DOI=10.1145/2414536.2414591 (2012)
8. Magnusson, C., Molina, M., Rassmus-Gröhn, K., Szymczak, D.: Pointing for non-visual orientation and navigation. In: NordiCHI 2010, pp. 735–738. ACM (2010)
9. Poppinga, B., Magnusson, C., Pielot, M., Rassmus-Gröhn, K.: TouchOver Map: Audio-Tactile Exploration of Interactive Maps. In: MobileHCI'11, pp. 545-550 (2011)
10. Qian, H., Kuber, R., Sears, A.: Towards identifying distinguishable tactons for use with mobile devices. In: 11th International ASSETS '09, pp. 257-258. ACM, New York, USA. http://doi.acm.org/10.1145/1639642.1639703
11. Sadato, N., Pascual-Leone, A., Grafmani, J., Ibanez, V, Deiber, MP, Dold, G, Hallet, M.: Activation of the primary visual cortex by Braille reading in blind. Nature 380, pp. 526 - 528, doi:10.1038/380526a0 (1996)
12. Yatani, K., Truong, K. N.: SemFeel: a user interface with semantic tactile feed-back for mobile touch-screen devices. In: 22nd UIST '09, pp. 111-120. ACM, New York, USA http://doi.acm.org/10.1145/1622176.1622198 (2009)