OS-Level Surface Haptics for Touch-Screen Accessibility

Suhong Jin, Joe Mullenbach, Craig Shultz, J. Edward Colgate, Anne Marie Piper
Northwestern University
2145 Sheridan Road
Evanston, IL 60208
{suhongjin, mullenbach, craigdshultz}@u.northwestern.edu, {colgate, ampiper}@northwestern.edu

ABSTRACT

The TPad Tablet combines an Android tablet with a variable friction haptic touch-screen and offers many novel interaction possibilities. For example, unique textures may be associated with different user interface elements, such as text boxes and buttons. This paper presents an Android AccessibilityService that was created to give operating system-wide (OS) access to haptic effects. Prior to this work, the haptic feedback of the TPad could be controlled only from within specific applications. With the new implementation, all applications and primary user interfaces (e.g. home screen) will have access to the TPad. Rather than focus on specific elements or applications, we seek to provide a high fidelity haptic experience that elevates the TPad's accessibility to the standard of Talkback and Voiceover, Android's and Apple's accessibility programs respectively. The code for the application is available on our website.

Categories and Subject Descriptors

H.5.2 [Information Interfaces And Presentation]: User Interfaces - Haptic I/O. K.4.2 [Social Issues]: Assistive Technologies For Persons With Disabilities.

Keywords

Surface Haptics, Touchscreen, Tablet, Accessibility, Variable Friction

1. INTRODUCTION

Existing accessibility solutions such as Apple's Voiceover and Android's Talkback services rely heavily on audio cues [3, 5]. Both programs speak to the user, providing information on the current screen and any UI element that a user touches. As has been previously reported, auditory feedback may not always be the optimal solution [11]. Users may not hear well in noisy environments and headphones can limit user's awareness of obstacles in their path and surroundings. With haptic feedback, a user can quickly identify different elements on the screen [11]. To realize this vision more broadly, we implemented OS-level haptic accessibility using the TPad Tablet, a variable friction haptic surface integrated with a 7-inch Android tablet shown in Figure 1 [8, 12]. Different than vibration feedback that is most common in mobile devices today, the TPad surface is driven ultrasonically to reduce friction between the surface and the fingertip. This creates a slippery feeling that is felt only when the fingertip slides across the screen, and not on the hand that is holding the device. Past versions of the TPad Tablet software allowed haptic feedback to be accessed through specific applications that called TPad methods. In this project, an operating system-level interface to haptic functionality was created through Android's AccessibilityService.

2. Background

In recent years, several attempts have been made to improve touch-screen accessibility for people with vision impairments. Touchplates [10] gave users different acrylic overlays, including a QWERTY keyboard and a numeric keypad to provide tactile feedback during typing. Other overlays include a map, mouse, and a menu bar. While a useful solution, the user is constrained to a physical haptic guide, limiting its flexibility and requiring them to manage physical plates in addition to their device. Haptic rendering of images has also been researched to allow devices to provide haptic feedback on 2d images [6, 9]. By analyzing the colors of the images, various textures were mapped to the image. Electrovibration haptic feedback has also been used in a mobile, visuo-tactile sensory device for the visually impaired [1]. The device has an attached webcam used to explore the user's surroundings and identify an object of interest. The object is displayed on a screen at which point a finger can be swept across the screen, in essence feeling the object.

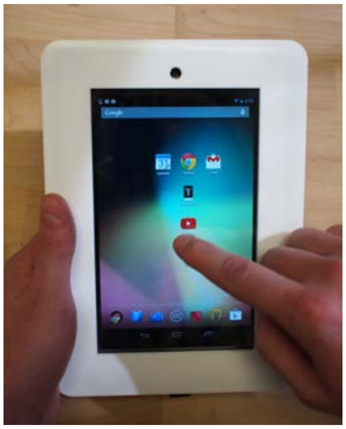

Figure 1: TPad in use on the home screen

Other techniques focus on novel gesture-based input to improve accessibility of number entry (e.g. DigiTaps [11]). In this system, the user performs a unique combination of swipes and taps to input a specific digit. The system provides haptic feedback to notify the user that the gesture was received and correctly interpreted. Audio feedback is also optionally available.

3. Implementation

In the present work, rather than focus on specific elements or applications, we seek to provide high fidelity haptic experience that increases accessibility across the entire device, much as Talkback and Voiceover do for audio. We use a time based friction modulation method to create distinct "texture tones", single-

Conference'10, Month 1–2, 2010, City, State, Country.

frequency amplitude modulated variable friction waveforms, for different icons and elements on the screen. This method of variable friction rendering has been used in previous studies, and was found to have a range of expressive capabilities, interpreted by users as a physical texture, as a sensation, or as an action [7].

3.1 AccessibilityService

An AccessibilityService is a background service that can be installed on an Android device and turned on through the Accessibility section of the settings of the smart device [2]. Talkback is one example of an AccessibilityService [5]. The AccessibilityService detects when the user interface's state changes. When the user puts his or her finger down on the screen (Figure 1), the service alerts the system of the touch interaction. If the user enters a view (for example, a button) the service also detects that change. As the users explores different UI elements, any changes are sent to the service to be handled.

3.2 Integrating the TPad

The TPad communicates with Android applications through IOIO libraries [4]. The IOIO libraries specific to Android applications have been modified to generate a custom library. When a developer creates an Android application, he or she adds this custom library to use the TPad-specific methods. For the AccessibilityService, a IOIO class that worked with services, instead of applications was created. The front-end TPadAccessibilityService class utilizes this new IOIO class to access the TPad.

3.3 Handling AccessibilityEvents

All AccessibilityService classes require an onAccessiblityEvent method. This method is called when an AccessibilityEvent is triggered. The TPadAccessibilityService focuses primarily on two AccessibilityEvents. The VIEW_HOVER_ENTER event occurs when a user slides his or her finger over a UI element, while VIEW_HOVER_EXIT is triggered when exiting a UI element. When the user's finger moves to a UI element, the TPad sends a texture tone while exiting the element turns the TPad off. Information about the UI element examined can be retrieved through AccessibilityRecords [2]. For example, the service can retrieve the contentDescription tag of elements, a concept Android documentation has stressed when implementing accessibility for applications. In the current implementation of the service, a look-up table of a few popular applications is searched with the UI element's currentDescription, though this could easily be expanded and customized to cover all applications. If the string is in the table, a unique texture is generated. Otherwise, the service checks the view's properties (editable, checkable, clickable, etc.). Each of these different properties have different texture tones to distinguish them from each other.

4. Advantages of AccessibilityService

With an operating system-level implementation of the TPad, Android application developers need not add TPad-specific code. The AccessibilityService automatically generates TPad texture tones through the service. However, developers can still add additional haptic feedback by implementing TPad methods within their application. Using the AccessibilityService also allows the TPad to work on the home screen, and other native applications. The AccessibilityService can also allow Android developers to customize a texture tone for a given element in their application. The developer can pass a Bundle with the properties of a TPad texture (floats for frequency and amplitude and an integer for wave type). The AccessibilityService can retrieve the Bundle through the AccessibilityRecord and generate the custom texture.

The ability to generate specific textures based on whether a view is editable, checkable, clickable, etc. elevates the TPad Tablet's accessibility toward the standard of Voiceover & Talkback. A user can be on any screen in the smart device, whether it's the home screen, the browser, or a specific application, and detect buttons, checkmarks, and edit boxes not just with audio and vibrational feedback from Voiceover & Talkback, but also with variable friction surface haptic feedback from the TPad.

5. Conclusion

Android's AccessibilityService has been used to give developers operating system-wide access to haptic feedback via the TPad. This makes the TPad available not just in specific applications, but in the home screen and other native application pages and should pave the way to broader and more diverse uses of surface haptic technology. The code is open-source and available for download [12].

6. ACKNOWLEDGMENTS

This work was supported by NSF Grant 0964075, McCormick School of Engineering, and the Segal Design Institute. Thanks also to the open-source IOIO project [4]. Special thanks to the many contributors to The TPad Tablet Project [12].

7. REFERENCES

[1] Ali Israr, Olivier Bau, Seung-Chan Kim, and Ivan Poupyrev. 2012. Tactile feedback on flat surfaces for the visually impaired. New York, NY, USA, 1571-1576.

[2] Android Accessibility. http://developer.android.com/guide/topics/ui/accessibility/index.html. Acc. June 17, 2014.

[3] Apple Inc., iPhone Accessibility. http://www.apple.com/accessibility/ios/. Acc. June 16, 2014.

[4] Ben-Tsvi, Y. IOIO Documentation. https://github.com/ytai/ioio/wiki. Acc. June 17, 2014.

[5] Google, Android Accessibility Help Center. http://www.support.google.com/talkback/. Acc. June 16, 2014.

[6] Jialu Li, Aiguo Song, and Xiaorui Zhang. 2010. Image-based haptic texture rendering. New York, NY.

[7] Joe Mullenbach, Craig Shultz, J. Edward Colgate, Anne Marie Piper. 2014. Exploring Affective Communication Through Variable-Friction Surface Haptics. (CHI '14). New York, NY.

[8] Joe Mullenbach, Craig Shultz, Anne Marie Piper, Michael Peshkin, and J. Edward Colgate. 2013. Surface haptic interactions with a TPad tablet. (UIST '13 Adjunct). New York, NY.

[9] Seung-Chan Kim, Ki-Uk Kyung, Dong-Soo Kwon. 2011. Haptic annotation for an interactive image. New York, NY.

[10] Shaun K. Kane, Meredith Ringel Morris, and Jacob O. Wobbrock. 2013. Touchplates: low-cost tactile overlays for visually impaired touch screen users. New York, NY.

[11] Shiri Azenkot, Cynthia L. Bennett, and Richard E. Ladner. 2013. DigiTaps: eyes-free number entry on touchscreens with minimal audio feedback. New York, NY.

[12] TPad Tablet Project. http://www.tpadtablet.org. Acc. June 17, 2014.