Touchpad Mapper: Exploring Non-Visual Touchpad Interactions for Screen-Reader Users

Ather Sharif*
asharif@cs.washington.edu
Paul G. Allen School of Computer Science & Engineering | DUB Group, University of Washington
Seattle, Washington, USA

Venkatesh Potluri*
vpotluri@cs.washington.edu
Paul G. Allen School of Computer Science & Engineering | CREATE, University of Washington
Seattle, Washington, USA

Jazz R. X. Ang
jazz.angruixia@gmail.com
Human-Centered Design & Engineering | DUB Group, University of Washington
Seattle, Washington, USA

Jacob O. Wobbrock
wobbrock@uw.edu
The Information School | DUB Group, University of Washington
Seattle, Washington, USA

Jennifer Mankoff
Paul G. Allen School of Computer Science & Engineering | CREATE, University of Washington
Seattle, Washington, USA

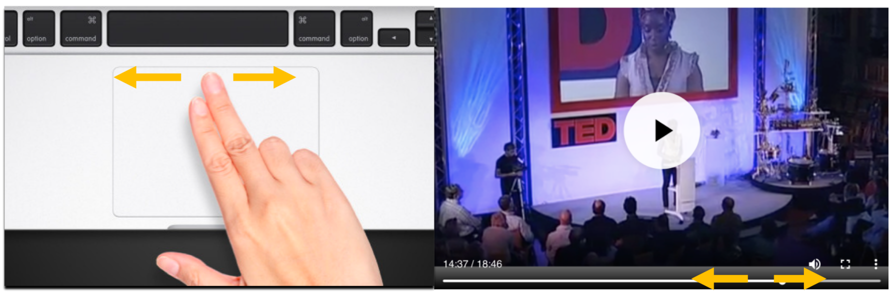

Figure 1: Using the touchpad to control a video slider (Context 1).

ABSTRACT

Despite their widespread availability, touchpads provide minimal utility for screen-reader users. We demonstrate the utility of these touchpads as input devices for screen-reader users through the development and preliminary evaluation of Touchpad Mapper, a system that maps digital content (e.g., image and video) to the physical coordinates of a touchpad. We explored two usage scenarios: (1) identification of objects and their relative positioning in an image, and (2) manipulation of a video seek bar or slider with rewinding and fast-forwarding features. Our pilot with three screen reader users shows that our participants expressed they extracted information faster using our system than a conventional keyboard.

KEYWORDS

touchpad, screen reader, blind, interaction, non-visual, video

*These authors contributed equally to this work.

W4A '24, May 13–14, 2024, Singapore

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

ACM Reference Format:
Ather Sharif, Venkatesh Potluri, Jazz R. X. Ang, Jacob O. Wobbrock, and Jennifer Mankoff. 2024. Touchpad Mapper: Exploring Non-Visual Touchpad Interactions for Screen-Reader Users. In *W4A '24: ACM International Web for All Conference, May 13–14, 2024, Singapore.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Touchpads have existed as part of laptop computers for several decades and provide utility to users through multi-touch gesture support, input capabilities and vibrotactile feedback. However, this widely used hardware offers minimal utility to screen-reader users for meaningfully interacting with digital content. Therefore, the digital interactions of screen-reader users remain predominantly linear and through discrete input devices, such as keyboards.

Touch interaction has become a standard on smartphones since first described by Kane *et al.* [2]. Furthermore, exploratory efforts [1, 3, 4] have demonstrated the potential of non-linear navigation in desktop screen readers. JAWS, NVDA, and VoiceOver, among other screen readers, support exploring the interface by mousing (similar to smartphone screen reader interactions). However, this mostly replicates interactions that are efficient when performed with a keyboard. In contrast, Touchpad Mapper demonstrates the *untapped* potential of touchpads for mapping 2D digital content, such as images and videos, to the physical layout of the touchpad.

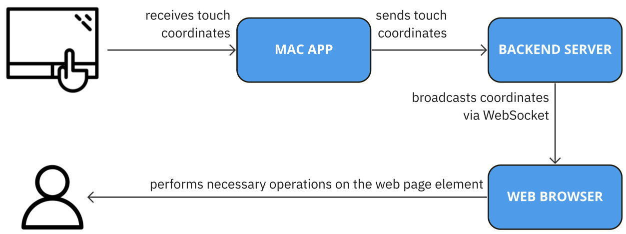

Figure 2: System diagram of Touchpad Mapper.

Figure 3: Image containing several landmarks (Scenario 2).

To our knowledge, Touchpad Mapper is the first demonstration of touchpad-enabled interaction techniques for screen-reader users.

We explore two usage scenarios for Touchpad Mapper. We map an image to the physical dimensions of a touchpad using absolute coordinates, enabling the identification of objects and their relative positioning in the image through finger-touch navigation. We also enable users to control a video slider and to rewind or fast-forward the video using the touchpad.

We assessed Touchpad Mapper through pilot studies with three screen-reader users, examining the utility of our system's interactions. Participants found our system useful and faster than a conventional keyboard interaction with digital content. They provided shared their positive experiences and suggestions for improvement.

2 TOUCHPAD MAPPER

We present Touchpad Mapper, its use scenarios for images and videos, and its system design (see Figure 2).

2.0.1 Scenario 1: Images. In this scenario, we mapped an image to the dimensions of the touchpad, assisting users to identify objects in the image by moving their finger within the physical bounds of the touchpad, similar to the interaction Slide Rule enables on smartphones [2]. We used Google's Vision API[1] to identify objects and their locations within the image on page load. When a user moves over a recognized object in an image, Touchpad Mapper reads out the object aloud. For example, in Figure 3, a user could locate landmarks, such as "Eiffel Tower," and its relative position.

2.0.2 Scenario 2: Videos. For this scenario, we mapped the length of a video slider to the top horizontal edge of the touchpad (see Figure 1), enabling users to alter the seek position. We used the bottom horizontal edge of the touchpad for rewind and fast-forward the video. Positioning the finger toward the left rewinds the video, whereas placing it toward the right fast-forwards the video.

2.1 Acquiring Touch Coordinates

To create Touchpad Mapper, we developed a MacOS App using Apple's AppKit and utilized the NSEvent[2] API to acquire touch coordinates and gestures from touchpad events. To send this information to the browser client, we created a WebSocket.[3] We developed a backend server using NodeJS to receive the coordinates from our App and broadcast this data to the user's browser.

[1]https://cloud.google.com/vision
[2]An object that contains information about an input action, such as a key press.
[3]A computer communications protocol that provides full-duplex communication channels over a single TCP connection.

2.2 Browser Client

To explore the utility of non-visual touchpad interactions, we created two web pages, one with an image and one with a video. We used JavaScript and the WebSocket API to receive touch coordinates from the backend server. Each page contained a client-side script to determine and perform actions based on the coordinates and relay information to the user with the SpeechSynthesis interface of the Web Speech API. For example, on the image page, we used the coordinates to announce the objects in the image.

3 PILOT STUDY

We conducted a pilot study with three blind screen-reader users to get feedback on Touchpad Mapper and investigate the utility of touchpad interactions for screen-reader users. We assigned participants the task to freely interact with our two usage scenarios (image and video) and subsequently interviewed them. They highlighted several beneficial use cases of using touchpads ranging from mathematical content to web page elements, especially images. They reported positive experiences interacting with Touchpad Mapper and considered it faster than a conventional keyboard interaction. Participants also provided us feedback on future improvements, such as enabling object detection to explore video frames.

4 CONCLUSION

We investigated the utility of a touchpad as an input device for screen-reader users. We developed Touchpad Mapper, a system that absolutely maps the dimensions of an image to the coordinates of a touchpad and enables image exploration via touch and object recognition. Touchpad Mapper also enables video exploration and playback control. We conducted a pilot study with three participants, finding that they reported Touchpad Mapper to offer useful, efficient, and promising interaction techniques.

ACKNOWLEDGMENTS

This work was supported by CREATE. We thank Matt Zeigler, Yisu Wang, James Fogarty, Jon Froehlich, and Richard Ladner for their comments and feedback. We remember our departed team member, Zoey. Without her feline support, the *purr*suit of this work would not have been as enjoyable. May she cross the rainbow bridge.

REFERENCES

[1] Mark S. Baldwin, Gillian R. Hayes, Oliver L. Haimson, Jennifer Mankoff, and Scott E. Hudson. 2017. The Tangible Desktop: A Multimodal Approach to Nonvisual Computing. *ACM Transactions on Accessible Computing (TACCESS)* 10, 3 (Nov. 2017), 9. https://doi.org/10.1145/3075222

[2] Shaun K. Kane, Jeffrey P. Bigham, and Jacob O. Wobbrock. 2008. 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*. ACM Press, Halifax, Nova Scotia, Canada, 73. https://doi.org/10.1145/1414471.1414487

[3] Rushil Khurana, Duncan McIsaac, Elliot Lockerman, and Jennifer Mankoff. 2018. Nonvisual Interaction Techniques at the Keyboard Surface. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*. ACM Press, Montreal QC, Canada, 1–12. https://doi.org/10.1145/3173574.3173585

[4] Ather Sharif, Olivia H. Wang, Alida T. Muongchan, Katharina Reinecke, and Jacob O. Wobbrock. 2022. VoxLens: Making Online Data Visualizations Accessible with an Interactive JavaScript Plug-In. In *CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 478, 19 pages. https://doi.org/10.1145/3491102.3517431