

A11yBoard: Using Multimodal Input and Output to Make Digital Artboards Accessible to Blind Users

Zhuohao Zhang

The Information School | DUB Group
University of Washington, Seattle, WA, USA
zhuohao@uw.edu

Jacob O. Wobbrock

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

ABSTRACT

We present A11yBoard, an interactive multimodal system that makes interpreting and authoring digital artboards, such as presentation slides or vector drawings, accessible to blind and low-vision (BLV) users. A11yBoard combines a web-based application with a mobile touch screen device such as a smartphone or tablet. The artboard is mirrored from the PC onto the touch screen, enabling spatial exploration of the artboard via touch and gesture. In addition, speech recognition and non-speech audio are used for input and output, respectively. Finally, keyboard input is used with a custom search-driven command line interface to access various commands and properties. These modalities combine into a rich, accessible system in which artboard contents, such as shapes, lines, text boxes, and images, can be interpreted, generated, and manipulated with ease. With A11yBoard, BLV users can not only consume accessible content, but create their own as well.

CCS CONCEPTS

• **Human-centered computing** → **Human computer interaction (HCI); Accessibility technologies.**

ACM Reference Format:

Zhuohao Zhang and Jacob O. Wobbrock. 2022. A11yBoard: Using Multimodal Input and Output to Make Digital Artboards Accessible to Blind Users. In *The Adjunct Publication of the 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22 Adjunct)*, October 29–November 2, 2022, Bend, OR, USA. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3526114.3558695>

1 INTRODUCTION

Digital object canvases are used in creativity software tools like Adobe Illustrator, Microsoft PowerPoint, Google Slides, and Apple Keynote. These canvases accept *objects* like shapes, text boxes, and images, and are sometimes called digital “artboards.” (By contrast, digital “paint canvases” accept pixels, like in Adobe Photoshop or Microsoft Paint.) For blind and low-vision (BLV) users, although extensive research has been conducted in delivering non-visual information like by using audio [2, 6, 12, 14] and haptics [7–9, 13], and by supporting accessible drawing [1, 3, 4, 8], most creativity software tools still remain inaccessible, especially object-based 2-D

digital artboards [10]. As the prevalence of digital artboards in professional and educational settings is high, making them accessible is of paramount importance.

Specifically, digital artboards present a variety of accessibility challenges [10], including high cognitive loads; difficulty determining object relationships, positions, and properties; and uncertainty about the success (or failure) of operations and commands, such as changing an object’s color or position relative to another object. It can be difficult for BLV users to build or maintain an accurate mental model of an artboard’s contents, to understand complex object relationships like object overlap (Z-order), objects’ relative positions, orientations, or sizes, or properties like color. A fundamental challenge is that conventional screen readers were developed primarily to handle text, which is a one-dimensional (1-D) information stream. But a digital artboard is inherently a 2-D information *space*, which is ill-suited to today’s screen readers. For example, the screen reader that ships with Microsoft Windows 10, when pointed at a Microsoft PowerPoint slide, announces the objects based on their Z-order, regardless of where these objects appear on the artboard. Why? Because the Z-ordering of objects is a well-defined 1-D information stream. (But it is hardly sufficient for the non-visual understanding of a 2-D artboard’s contents.)

To address this problem, we present A11yBoard, an interactive multimodal system that provides accessible interpreting and authoring of digital artboards. A11yBoard combines a web-based application with a mobile touch screen device such as a smartphone or tablet (Figure 1). The artboard is mirrored from the PC onto the touch screen, enabling spatial exploration of the artboard via touch and gesture, an approach adapted from prior work on finger-driven screen reading [5]. In addition, speech recognition and non-speech audio are used for input and output, respectively. Finally, the keyboard is used with a custom search-driven command line interface to access various commands and properties. The result is a holistic user experience that enables BLV users to both consume and create digital content. In the sections that follow, we discuss A11yBoard’s interface and features, followed by three pilot studies and plans for a more formal user evaluation.

2 THE A11YBOARD SYSTEM

A11yBoard is a combined multimodal system comprising a web-based interface on desktop PC and a mobile touch screen interface. The web interface side is built upon a web-based open-sourced canvas drawing tool [11], which we modified substantially. This drawing tool is a JavaScript WYSIWYG HTML canvas editor consisting of basic object-based drawing functions including inserting shapes, lines, text boxes, and images. Objects have properties like position, size, orientation, and color, which can be inspected and

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

UIST '22 Adjunct, October 29–November 2, 2022, Bend, OR, USA

© 2022 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-9321-8/22/10.

<https://doi.org/10.1145/3526114.3558695>

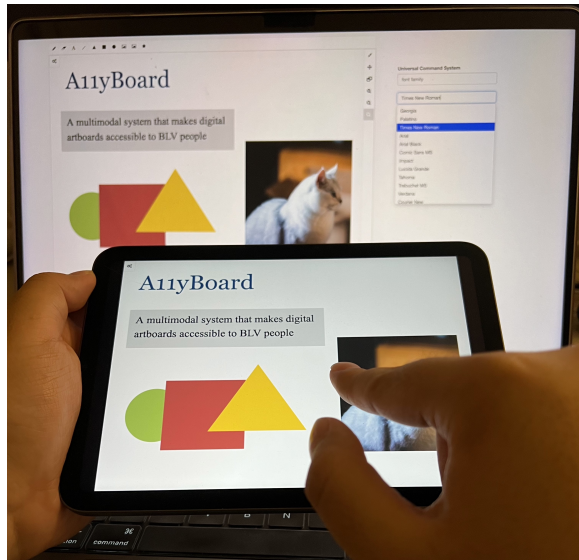


Figure 1: Using A11yBoard, a web-based drawing tool with a mirrored touch display enabling risk-free exploration and audio interactions. A11yBoard’s supports interpreting via touch and gestures, and outputs information about objects using audio tones and speech. For example, when a user’s finger crosses into an object, a “step up” tone is played. When a user split-taps an object, speech output describes it. When a user’s finger crosses out of an object, a “step down” tone is played. A11yBoard also supports generating and modifying artboard contents using speech or keyboard commands.

modified. Drawing functions are similar to existing artboard tools like Microsoft PowerPoint or Google Slides.

As noted, A11yBoard employs a variety of multimodal inputs and outputs across three pieces of hardware (desktop display, touch screen device, and desktop keyboard; the mouse is intentionally avoided). The desktop display shows the artboard. The touch screen device mirrors that artboard and allows for risk-free finger-driven spatial exploration of the artboard’s contents. It also provides audio output, for example, when the finger crosses into and out of objects, and speech input, for example, when asking for the nearest object to a location. The desktop keyboard can be used with what we call a Universal Command Line, where the user can simply type a few characters related to their need and a search procedure will provide an enumerated list of most likely commands and property values (e.g., “mag” for setting an object to the color “magenta”). Whereas the touch screen allows users to spatially explore an artboard’s contents through continuous input (i.e., finger movement), the keyboard allows for execution of discrete commands, such as creating new objects (e.g., “rec” for creating a new rectangle) or changing properties (e.g., “fro” for bringing an object to the front of the Z-order). We elaborate on key features below.

2.1 Interpreting Artboards via Touch and Gesture

A11yBoard uses a touch screen to mirror the desktop’s artboard and enable risk-free exploration via finger-driven screen reading similar to that pioneered by Slide Rule [5]. By mirroring the desktop’s artboard onto a touch screen device, the artboard can be explored *spatially* in a 2-D fashion, giving BLV users a notion of objects’ relative positions and sizes. Moving with a finger over objects results in basic auditory feedback, while simple gestures give more detailed object information on-demand.

Single-finger exploration: A11yBoard supports a single finger “reading” the artboard’s contents by moving over the touch screen. When a user’s finger moves into an object, A11yBoard plays a “step-up” tone; similarly, when a user’s finger moves out of an object, A11yBoard plays a “step-down” tone. When transitioning between overlapping objects, A11yBoard plays these tones in quick succession. If the finger moves slowly enough so as to remain within an object momentarily, A11yBoard reports the basic information about the object, including its color name and type (e.g., “green rectangle”). Any speech output is abruptly cut-off if the finger exits the object or lifts from the touch screen.

Split-tap drill-down: While exploring the canvas using one finger, users may want to access detailed information about objects, or select objects for further interaction. To this end, A11yBoard supports split-tap (a second-finger tap issued while the first finger remains down [5]) to select an object and hear detailed information about it, including its size and location, which are reported using relative ratios of the canvas width and height. For collaboration with sighted or low-vision users, A11yBoard also visually highlights the selected object with a thicker border.

Single-finger dwell: To further interact with a selected object, A11yBoard supports a variety of speech commands, which will be actively listened for when one finger dwells on the screen. Speech interaction will stay active as long as the finger dwells, much like the Shift key on a keyboard. We further describe A11yBoard’s speech commands in section 2.2, below.

Quick-flick to nearby objects: To support learning of surrounding objects, A11yBoard offers a two-finger quick-flick interaction that reads out information about the closest object in the flicking direction. By using these flicks, a user can explore “radially” around an object to understand nearby objects in various directions.

2.2 Speech-Based Interaction

When BLV users hold down a finger on the screen to “talk” with A11yBoard, the system supports a series of speech commands to access information and manipulate objects. Holding down a finger is an active mode, much like the Shift key on a keyboard. It is also similar to holding down the button on a walkie talkie when speaking. While the user’s finger dwells, A11yBoard is “listening.” When the user’s finger lifts, speech mode ceases.

The speech commands used to retrieve detailed object information include “position” (and related commands like “left”, “top”, “right”, “bottom”), “color”, “size” (and related “width”, “height”), “closest”, “farthest” (reads out the closest or farthest object from the finger’s position), and “text” (reads out the text object’s inner text).

The reported results of “position” and “size” will be percentage numbers proportional to the canvas width and height. For example, “the width is 25% of the canvas width.” If users want more precise values, these commands can be extended by appending the word “exact.” For example, “position exact” might result in “red rectangle from canvas left: 145 pixels.”

Speech commands can also be used to create or edit objects, including “create”, “move”, “resize”, “align”, “here”, and “cancel.” “Create” is followed with an object type like “rectangle” or “text” and results in a new object under the user’s dwelling finger. “Move” and “resize” are used to initialize a moving or resizing operation. “Resize” is specified with a object reference point, like “top-left,” which would be the corner from which the resizing occurs. Users can then continue exploring the canvas until they find their desired moving or resizing destination, and then use command “align” or “here” to either align the object to another object’s edge or to simply move or resize object to the new location.

2.3 Universal Command Line

Touch, gesture, and speech commands are powerful means for BLV users to interact with 2-D information spaces like artboards, but they are still limited when it comes to certain functions like changing discrete values of color, font family, and font size. To make nearly all A11yBoard functions easily and quickly discoverable in one location, we created what we call the “Universal Command Line.” This command line is a search-driven keyboard interface enabled by speech output into which BLV users can type keywords or command terms like “color” and “font,” and then select from existing results appearing in a drop-down menu, which is made accessible by text-to-speech output. Additional commands like “copy,” “delete,” “bring-to-front,” and “send-to-back” are also supported for further editing the canvas. Importantly, because the command line is search-driven, users do not need to memorize all possible commands. Rather, they only need to type a few characters of a desired term, and the search feature will present all possibilities to them.

3 PILOT STUDY AND FUTURE WORK

To gather feedback about A11yBoard, we conducted a pilot study with three blind people who had experience making or at least using presentation slides, which are a type of artboard. We asked participants to use A11yBoard to interpret an existing artboard and to edit objects within it. We first demonstrated A11yBoard’s features and asked for participants’ feedback. We then gave participants various tasks. In general, participants were extremely positive about A11yBoard, all enjoying using it to access the given artboard and using interaction techniques to edit it. Participants were able to tell us accurately what an artboard generally contained (e.g., a slide with a title in the top-left corner, a rectangle below that title, and a few other shapes on the right side). One participant appreciated the design of the “move” and “resize” features because they allowed him to explore the “destination” first and then decide to conduct the moving or resizing operation. Participants also gave a few suggestions on how to improve the system, including the need of adding a summary overview of the entire artboard. (We plan to add a “summarize” command as a result.) All three participants wanted this summary before exploring the artboard using touch

and gesture in a risk-free way. Participants also wanted to have a “help” speech command to remind them of the possible interaction techniques and speech commands.

Our pilot study results indicate the usefulness of combining touch, gesture, speech, audio, and search for improving digital artboard accessibility. In our future work, we will improve and extend the design of A11yBoard, for example, by adding “summarize” and “help” commands. We will also perform additional iterative usability tests with more BLV people. On the whole, we believe A11yBoard will be a significant step towards making 2-D information spaces like digital artboards more accessible, a crucial step towards empowering BLV users to create design artifacts of their own.

ACKNOWLEDGMENTS

This work was supported in part by the University of Washington Center for Research and Education on Accessible Technology and Experiences (CREATE). Any opinions, findings, conclusions or recommendations expressed in our work are those of the authors and do not necessarily reflect those of any supporter.

REFERENCES

- [1] Jens Bornschein and Gerhard Weber. 2017. Digital Drawing Tools for Blind Users: A State-of-the-Art and Requirement Analysis. In *Proceedings of the 10th International Conference on Pervasive Technologies Related to Assistive Environments (PETRA '17)*. Association for Computing Machinery, New York, NY, USA, 21–28. <https://doi.org/10.1145/3056540.3056542>
- [2] L. M. Brown, S. A. Brewster, S. A. Ramloll, R. Burton, and B. Riedel. 2003. Design guidelines for audio presentation of graphs and tables. <https://eprints.gla.ac.uk/3196/>
- [3] William Grussenmeyer and Eelke Folmer. 2016. AudioDraw: user preferences in non-visual diagram drawing for touchscreens. In *Proceedings of the 13th International Web for All Conference (W4A '16)*. Association for Computing Machinery, New York, NY, USA, 1–8. <https://doi.org/10.1145/2899475.2899483>
- [4] Hesham M. Kamel and James A. Landay. 2000. A study of blind drawing practice: creating graphical information without the visual channel. In *Proceedings of the fourth international ACM conference on Assistive technologies (Assets '00)*. Association for Computing Machinery, New York, NY, USA, 34–41. <https://doi.org/10.1145/354324.354334>
- [5] 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)*. Association for Computing Machinery, New York, NY, USA, 73–80. <https://doi.org/10.1145/1414471.1414487>
- [6] Shaun K. Kane, Meredith Ringel Morris, Annuska Z. Perkins, Daniel Wigdor, Richard E. Ladner, and Jacob O. Wobbrock. 2011. Access overlays: improving non-visual access to large touch screens for blind users. In *Proceedings of the 24th annual ACM symposium on User interface software and technology - UIST '11*. ACM Press, Santa Barbara, California, USA, 273. <https://doi.org/10.1145/2047196.2047232>
- [7] Shaun K. Kane, Meredith Ringel Morris, and Jacob O. Wobbrock. 2013. Touchplates: low-cost tactile overlays for visually impaired touch screen users. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '13)*. Association for Computing Machinery, New York, NY, USA, 1–8. <https://doi.org/10.1145/2513383.2513442>
- [8] Martin Kurze. 1996. TDraw: a computer-based tactile drawing tool for blind people. In *Proceedings of the second annual ACM conference on Assistive technologies (Assets '96)*. Association for Computing Machinery, New York, NY, USA, 131–138. <https://doi.org/10.1145/228347.228368>
- [9] Denise Prescher, Gerhard Weber, and Martin Spindler. 2010. A tactile windowing system for blind users. In *Proceedings of the 12th international ACM SIGACCESS conference on Computers and accessibility (ASSETS '10)*. Association for Computing Machinery, New York, NY, USA, 91–98. <https://doi.org/10.1145/1878803.1878821>
- [10] Anastasia Schaadhardt, Alexis Hiniker, and Jacob O Wobbrock. 2021. Understanding Blind Screen-Reader Users’ Experiences of Digital Artboards. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–19.
- [11] Carsten Schäfer. 2022. DrawerJS: a platform-independent, web-based JavaScript WYSIWYG HTML canvas editor. <https://www.drawerjs.com/>
- [12] Ather Sharif and Babak Fouraghi. 2018. evoGraphs – A jQuery plugin to create web accessible graphs. In *2018 15th IEEE Annual Consumer Communications &*

- Networking Conference (CCNC)*. 1–4. <https://doi.org/10.1109/CCNC.2018.8319239>
ISSN: 2331-9860.
- [13] Steven Wall and Stephen Brewster. 2006. Feeling what you hear: tactile feedback for navigation of audio graphs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. Association for Computing Machinery, New York, NY, USA, 1123–1132. <https://doi.org/10.1145/1124772.1124941>
- [14] Tsubasa Yoshida, Kris M. Kitani, Hideki Koike, Serge Belongie, and Kevin Schlei. 2011. EdgeSonic: image feature sonification for the visually impaired. In *Proceedings of the 2nd Augmented Human International Conference (AH '11)*. Association for Computing Machinery, New York, NY, USA, 1–4. <https://doi.org/10.1145/1959826.1959837>