Access Overlays: Improving Non-Visual Access to Large Touch Screens for Blind Users

Shaun K. Kane[1,6], Meredith Ringel Morris[2], Annuska Z. Perkins[3],
Daniel Wigdor[4], Richard E. Ladner[5], Jacob O. Wobbrock[6]

[1]Information Systems, UMBC
Baltimore, MD, 21250 USA
skane@umbc.edu

[2]Microsoft Research
Redmond, WA, 98052 USA
merrie@microsoft.com

[3]Microsoft Research
Redmond, WA, 98052 USA
annuskap@microsoft.com

[4]Computer Science
University of Toronto
Toronto, ON, Canada
dwigdor@dgp.toronto.edu

[5]Computer Science & Engineering
University of Washington
Seattle, WA 98195 USA
ladner@cs.washington.edu

[6]Information School | DUB Group
University of Washington
Seattle, WA 98195 USA
wobbrock@uw.edu

ABSTRACT

Many touch screens remain inaccessible to blind users, and those approaches to providing access that do exist offer minimal support for interacting with large touch screens or spatial data. In this paper, we introduce a set of three software-based *access overlays* intended to improve the accessibility of large touch screen interfaces, specifically interactive tabletops. Our access overlays are called *edge projection, neighborhood browsing,* and *touch-and-speak*. In a user study, 14 blind users compared access overlays to an implementation of Apple's VoiceOver screen reader. Our results show that two of our techniques were faster than VoiceOver, that participants correctly answered more questions about the screen's layout using our techniques, and that participants overwhelmingly preferred our techniques. We developed several applications demonstrating the use of access overlays, including an accessible map kiosk and an accessible board game.

Author Keywords

Accessibility, touch screens, blindness, visual impairments.

ACM Classification Keywords

H.5.2. Information interfaces and presentation: User Interfaces–*input devices and strategies.* K.4.2. Computers and Society: Social issues–*assistive technologies for persons with disabilities.*

General Terms

Design, Experimentation, Human Factors.

INTRODUCTION

For blind people, accessing touch screen interfaces remains a significant challenge, as most touch screens rely on visual interaction and are not usable by touch and audio alone.

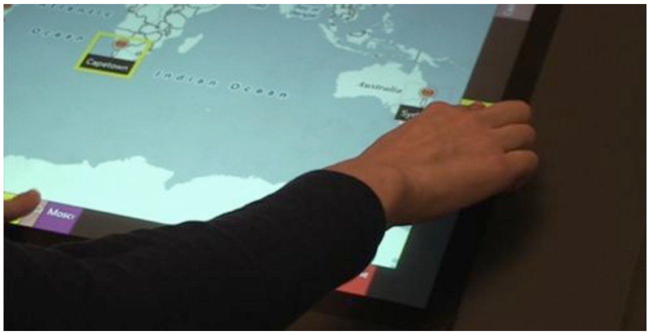


Figure 1. Access overlays allow a blind user to accurately locate items on a 2-D touch screen map.

Despite this significant problem, touch screens are increasingly found in consumer technologies such as mobile devices, home electronics, and computers in public spaces (*e.g.*, ATMs, airport ticket kiosks, and interactive maps). The inaccessibility of touch screens can have profound effects, preventing otherwise independent blind people from performing routine tasks without help, which can lead to feelings of embarrassment [17]. Blind consumers have responded to the spread of inaccessible technologies through press events [7], and have organized lawsuits and boycotts [23]. Furthermore, inaccessible touch screens not only impact millions of blind people (more than 1.3 million in the U.S. alone[1]), but also seniors and others with low vision, as well as other people who use touch screens eyes-free, such as while multitasking.

Fortunately, some touch screen devices now provide accessibility features for blind people. Many touch screen ATMs provide an alternative button-based interface and a headphone jack for blind users. In recent years, several research projects (*e.g.,* [10,16]) and commercial tools such as Apple's VoiceOver for iOS[2] and Google's Eyes-Free

UIST'11, October 16–19, 2011, Santa Barbara, CA, USA.

---

[1] http://www.nfb.org/nfb/blindness_statistics.asp

[2] http://www.apple.com/accessibility/iphone

Shell for Android[3] have introduced accessible touch interfaces that combine gesture input with speech output.

While techniques such as VoiceOver demonstrate that touch screens can be made at least somewhat accessible, it is not at all clear that such interfaces are optimal, as performance comparisons of accessible touch screen interfaces are rare. Furthermore, as larger touch screen devices become more common, additional interaction challenges emerge. In particular, increased screen size may result in increased search time to find on-screen targets, and many interactions with large touch screens may require a blind user to understand the spatial layout of the screen, such as when exploring a map or working collaboratively with a sighted partner. Most current touch screen accessibility tools are designed for small, mobile phone-sized devices, and may not address the challenges of interacting with larger touch screens, such as interacting with two hands and working with spatial data.

In this paper, we introduce *access overlays* (Figure 1), a set of new techniques that enable blind people to explore and interact with applications on interactive tabletops. We describe our formative work in developing these techniques, and introduce three access overlays. We then describe a study in which 14 blind computer users compared access overlays to VoiceOver. Our results show that access overlays enable users to perform tasks faster than VoiceOver, improve users' ability to answer questions about the screen's spatial layout, and are preferred.

RELATED WORK

Our work extends prior access techniques for small touch screens by introducing techniques for spatial exploration of larger touch screens, and is inspired by prior approaches to increase the accessibility of virtual maps and diagrams.

Touch Screen Accessibility

Touch screen interfaces have been popular for over 20 years [5], and concerns about touch screen accessibility have remained active throughout this time [4]. Touch screen accessibility research has considered various device form factors. Early research explored accessibility for devices such as information kiosks [29] and drawing tablets [19]. These techniques relied upon hardware modifications, such as augmenting a touch screen with physical buttons or placing a physical overlay atop the screen. Such approaches may be expensive to install, may limit the flexibility of the underlying software (by imposing physical structures), and may interfere with use by sighted people; unsurprisingly, such techniques have not been widely adopted.

More recent efforts have focused on using gestures to provide blind users with access to touch screens without modifying the underlying hardware. For example, Slide Rule [16] used multi-touch gestures to allow blind users to browse and explore content on a touch screen-based smartphone. Similar interfaces now appear in consumer devices such as Apple's iPhone[2] and Android-based smartphones[3]. Researchers have also explored techniques to address specific aspects of blind touch screen interaction, such as text entry [3,10,36] and gesture selection [18]. However, these techniques have often focused on small, mobile phone-sized screens. Furthermore, many of these techniques change the fundamental layout of the screen to improve accessibility. Thus, our current work addresses interaction with larger touch screens, and explores methods to preserve users' understanding of the screen layout.

Accessible Maps and Diagrams

While some touch screen accessibility techniques rely on changing the screen layout to improve usability, researchers have also explored methods to improve the accessibility of spatial information such as maps [20], walking directions [25,26], and diagrams [6,22,30]. However, these techniques have typically targeted specific domains, and are not generalizable to all touch screen interfaces. Furthermore, many of these techniques require haptic controllers or other specialized hardware. Our present work explores methods to access spatial layouts on unmodified multi-touch screens.

DESIGN GOALS FOR LARGE TOUCH SCREENS

When designing new touch screen interfaces for blind people, it is important to consider the criteria for success. Prior approaches have often considered performance metrics such as speed and error rate (*e.g.,* [3,16]). However, large touch screens such as tablets, kiosks, and interactive tabletops may introduce additional interaction challenges related to navigating spatial layouts, collaborating with others, and using unfamiliar devices in public spaces. Among those issues, our current work focuses on concerns related to *preserving spatial layout, leveraging bimanual interaction, reducing search space,* and maintaining usability in *walk-up-and-use scenarios.*

Preserving spatial layout. Access techniques that distort or remove spatial information may reduce users' spatial understanding and memory, making it more difficult for a blind person to understand maps and diagrams or to collaborate with sighted peers. Thus, blind users should be able to interact with the original spatial layout of an application when desired.

Leveraging bimanual interaction. Most accessible touch screen interfaces do not support bimanual interaction, even on larger touch screens. Accessible touch screen interfaces should leverage bimanual interactions on devices that support them to improve performance.

Reducing search space. Since blind people must search a touch screen using their hands, rather than glancing at the screen, the user interface should minimize the distance that a user needs to cover in order to search on-screen content.

Walk-up-and-use. Because touch screens are increasingly found in public spaces, touch screen interfaces for blind users should be usable in walk-up-and-use scenarios, and

---

[3] http://code.google.com/p/eyes-free

should not require users to memorize complex gestures or screen locations, or to recognize auditory icons.

In addition to the above criteria, there are additional challenges that are specific to larger touch screens, such as multi-user interaction and interacting with physical objects. Although our formal evaluation did not specifically address these issues, our techniques and applications were influenced by these concerns, and can be extended to address these concerns in the future.

FORMATIVE INTERVIEWS

We began our research by investigating how blind people organize their workspaces, to uncover organization strategies that could potentially be used to inform the design of touch screen user interfaces for blind people. We interviewed 8 blind people (4 female, 4 male, aged 31-66) in their place of work. The interview began with a guided tour of the participant's workspace, followed by a discussion of strategies used to organize and search that space. Finally, the participant was presented with a collection of souvenirs provided by the researcher (*e.g.,* postcards, figurines, and coins) and asked how he or she would organize those items to show them to a friend.

Organization strategies varied across participants. We inquired whether participants had received vocational training to help organize their workspace, but none had received any such training. Some participants had very rigorous organization structures, such as one person who "put everything in quadrants," while others were much less formal. However, participants' workspaces were typically organized from an egocentric viewpoint, such that objects were arrayed around the participant's chair. Frequently used objects were placed near the resting places of participants' hands, while less frequently used objects were moved to the corners of the workspace to keep them out of the way. Most participants noted the importance of keeping items in a consistent location so that they could be relocated later.

Search strategies also varied across participants. Most participants remembered the approximate location of an item, or the drawer it was in, but not its precise location. Participants would relocate the object by feeling around a general area for an item until they recognized it by touch. Because it was possible to accidentally bump or knock over an object while searching, participants were very careful to keep fragile objects in a consistent, out-of-the-way location, such as tucking a water cup behind a computer screen.

Our present design work was inspired primarily by our participants' use of structured spatial layouts in their workspaces, their use of edges and corners to organize space, and their use of local search to find items.

ACCESS OVERLAYS

Access overlays are accessible interaction techniques that improve touch screen usability while preserving an application's original spatial layout. Access overlays are implemented as semi-transparent windows that reside above a standard application. When activated, an access overlay gathers information about the location and content of all on-screen targets, and provides access to these targets through a combination of speech and audio feedback, alternative gesture input, and additional user interface controls. Access overlays are entirely software-based, and do not require alterations to the underlying touch screen hardware. While blind touch screen users would likely benefit from the additional haptic feedback provided by physical touch screen overlays, or by new technologies such as TeslaTouch [1], these technologies are not yet widely available. In contrast, access overlays are designed to improve the accessibility of existing touch screen hardware.

We developed a number of access overlays, and after iterative design and testing, refined three of the most promising overlays, which we now describe.

Edge Projection

One strategy used by prior touch screen access techniques (*e.g.,* [16,29]) is to convert a two-dimensional interface to a linear list of targets, which we call *linearization*. Linearization enables blind users to quickly scan a list of on-screen items without the need to search the entire screen, but removes information about the screen's original layout.

The *edge projection overlay* (Figure 2) provides the benefits of linearization while maintaining the original spatial layout of the screen. When edge projection is active, touching any on-screen target reads that target's name. In addition, the edge projection overlay displays an *edge menu* that surrounds the screen. Each on-screen target has a corresponding *edge proxy* along each edge of the screen. Touching the edge proxy highlights the corresponding on-screen target and reads its name. Blind users can quickly browse through the list of targets by sliding their finger across the edge menu. Furthermore, because the position of the edge proxy corresponds to the *x*- or *y*-position of the target, users can drag their fingers from an edge proxy toward the interior of the screen to locate the desired target.

Figure 2. Edge projection. Targets are projected to the edge of the screen. Lines illustrate the correspondence between an edge proxy and its associated target.

Edge projection leverages bimanual interaction by allowing users to locate on-screen targets using two hands. For example, the user may locate a target on the bottom edge with one hand, locate the same target on the right edge with another hand, and move their hands together toward the interior of the screen to locate the target. Our prototype also supports a *bimanual context menu* that allows the user to locate a target along one edge, and then browse a list of actions that can be performed on that target by touching a second edge. In another mode, the second edge can be used to select from a group of closely clustered targets: touching a cluster of targets along one edge causes the corresponding edge proxies to spread out along the second edge, allowing users to more easily select the desired proxy.

Edge projection was inspired by our interview participants' tendency to place objects in the edges and corners of their workspaces. Edge projection preserves the original layout of on-screen content, reduces search space, and leverages bimanual interaction. Edge projection also supports collaboration by allowing users to explore the screen using the edge closest to them, avoiding conflict with other users.

Neighborhood Browsing

A major difficulty in exploring a touch screen without sight is actually finding targets on the screen. Most visual interfaces use empty space to separate and group targets. Without appropriately designed feedback, a blind person touching an empty area of a touch screen might not know where they are touching, or even if the system has registered their touch. Locating targets on the screen likewise requires navigating through that empty space.

The *neighborhood browsing overlay* (Figure 3) addresses the problem of searching a large touch screen by increasing the size of targets and reclaiming empty space. Neighborhood browsing uses a Voronoi tessellation [8] to define a neighborhood around each on-screen target. Touching anywhere on the screen speaks the name of the nearest target. Users can precisely locate a target by touching within a target's neighborhood and performing a second-finger tap gesture [16]. The system then provides *guided directions* to the nearest target, as described in the *Additional Features* subsection below.

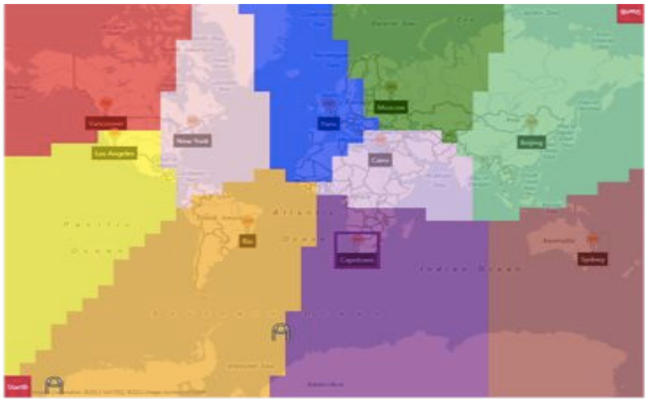


Figure 3. Neighborhood browsing uses a Voronoi tessellation to increase target size. Aliasing of the regions is a visual artifact, and did not affect functionality.

Neighborhood browsing was inspired by our participants' tendency to remember the approximate location of items in their workspace, and to search locally in that approximate location to find the item, and is also inspired by the bubble cursor [9]. Neighborhood browsing preserves spatial layout and reduces search space.

Touch-and-Speak

Blind PC users often rely on keyboard shortcuts that can be performed without sight. Because many touch screen-based devices do not have keyboards, blind touch screen users must enter commands by browsing through a speech menu, which may be slow and tedious. The *touch-and-speak overlay* allows users to perform actions much more quickly by combining touch interaction with spoken commands. Users initiate a voice command by performing a second-finger tap gesture on the screen and speaking a command.

Currently, three commands are supported: (1) saying *"list"* reads all on-screen targets from left to right; (2) saying *"nearby"* reads all targets in the same quadrant of the screen as the user's touch; and (3) speaking the name of a target provides guided directions from the user's finger to the named target. Because speech commands begin with a touch gesture, commands can be bound to the touch location, and recognition accuracy can be improved by only considering options relevant to the current location [27].

Touch-and-speak was inspired by our formative interviews and by prior work that used speech commands to improve pointing precision [27]. Touch-and-speak preserves spatial layout, and reduces search space by allowing users to find any target from any screen location. The simple voice commands also support walk-up-and-use scenarios.

Additional Access Overlays

We prototyped several other access overlays, including several types of area cursors [35], a *grid overlay* that divided the screen into a regular grid, and a *world-in-miniature overlay* that presented an overview of the entire screen in one corner. Pilot participants had difficulty using these overlays, and so we omitted them from our user study.

Additional Features

In addition to the access overlays described above, our prototype provides several other accessibility features.

Auditory Touch Feedback

Some touch screen applications present visual feedback to let users know that a touch event has been recognized [32]. Our techniques provide audible feedback of touch events, emitting a low beep whenever a touch is detected. This feedback helps to prevent errors in which the user touches the screen too lightly to be detected.

Audio Dividers

One advantage of using a large touch screen is the ability to divide the screen into sub-regions. In a visual interface, these regions are typically denoted by drawing borders or assigning different backgrounds. Our techniques use soft, looping background sounds to differentiate regions of the screen. This allows users to identify when they have moved between regions, and to identify the boundaries of a region.

Our current prototype uses white noise loops, filtered at different frequencies, to differentiate regions.

Guided Directions

Both neighborhood browsing and touch-and-speak offer guided directions from the user's current touch location to an on-screen target. Directions are created by an algorithm that generates straight line paths between the user's finger (x_u, y_u) and the on-screen target (x_t, y_t) as follows:

1. If $y_t < y_u$, move *down* to point (x_u, y_t);
2. If $x_t \neq x_u$, move *left* or *right* to x_t;
3. If $y_t > y_u$, move *up* to y_t.

Generated paths first move down (*i.e.,* toward the user) if possible, in order to keep the majority of movement close to the user, reducing the need to stretch and the likelihood of colliding with physical objects on the surface. Diagonal movements are avoided because pilot subjects found it difficult to accurately follow diagonal paths; however, participants were capable of following horizontal and vertical paths. If a physical object is present along the path, the system identifies its bounding box and uses *A** [12] to route around the object. Figure 4 shows an example path.

Figure 4. Path from the user's finger to an on-screen target. Directions are provided via speech feedback at each segment of the path, *e.g.*, "Right 8 inches."

Audio feedback is provided as spoken directions (up, down, left, and right), and distances (in inches). While some prior systems have used mappings between tone and direction to guide users (*e.g.,* [15,22]), spoken directions are usable without training, and are thus ideal for walk-up-and-use scenarios. The system first reads the direction to and distance of the next point in the path, *e.g.*, "Right 4 inches." As a user continues on the path, he or she receives continuous feedback to continue in that direction, *e.g.*, "Right, right, right, …" When the path changes direction, the system speaks the direction and distance to the next path point. If the user diverges from the path beyond a certain threshold, the system guides the user back onto the original path. All pilot testers and study participants were able to follow these directions effectively, and some participants could easily estimate distance, moving their finger to the next path point in one fluid movement, rather than relying upon continuous audio feedback.

EVALUATION

We conducted a user study of the three access overlays described above. The access overlays were compared to an implementation of Apple's VoiceOver, which is used by as many as 100,000 blind people [28]. The study examined both overall performance when using the interface, as well as participants' spatial understanding of the screen layout.

Participants

We recruited 14 blind computer users (7 male, 7 female), with an average age of 46.4 (SD=12.6). All participants used a screen reader. Five participants were current VoiceOver users, and two other participants had previously tried VoiceOver but did not regularly use a touch screen. No other participants regularly used a touch screen.

Apparatus

The study was conducted using a Microsoft Surface tabletop computer with a 24-inch by 18-inch multi-touch screen, running Windows Vista. Because the Microsoft Surface's touch screen was surrounded by a bezel that felt identical to the touch screen itself, we added electrical tape around the screen to make it easier for participants to find the edge. We also added an external microphone and speakers. No other hardware modifications were made.

Participants tested each technique using a custom application, referred to here as *grid-map* (Figure 5). This application presented a grid-aligned map containing random points of interest. Each grid cell was 2 inches square, and 5 items were shown on the map at any time. Of the 108 cells, 44 cells were excluded as they overlapped portions of the access overlays' user interfaces. To avoid confounds, points of interest were randomly placed, and used randomly generated names consisting of a place type ("Bar," "Café," "Hotel," or "Store") and a letter (*e.g.*, "Café Z").

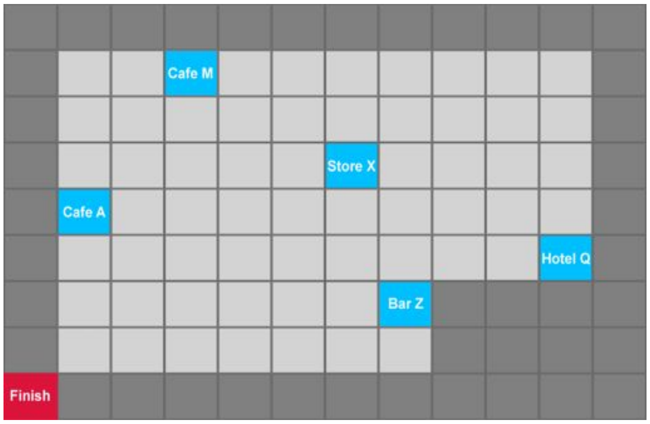


Figure 5. Grid-map application used in the experiment. Highlighted squares indicate points of interest on the map.

The study software recorded the start and end of each study trial, as well as the start time, end time, and position of each touch on the screen, calculated using the Surface SDK. Study data was saved to a log file. Participant feedback was recorded by the experimenter in a separate text file.

Procedure

Participants used each of the techniques to perform five distinct tasks using the grid-map application. To address both traditional performance metrics and spatial understanding, tasks involved both selecting targets and answering questions about the map's layout. The following tasks were used:

- *Locate.* Given the name of a target, the participant was required to touch the grid cell containing that target.
- *Count.* The participant was asked to report the number of targets on screen of a specified type (*e.g.*, hotels).
- *Relate.* The participant was given the names of two on-screen targets, and was asked to name the topmost or leftmost of the two targets (randomly chosen).
- *Select.* The participant selected a target, either by directly touching it or by some other method, such as by touching its edge proxy. When a target was selected, the system spoke the target name and a randomly selected "opening time." The participant was asked to report the opening time.
- *Relocate.* This task consisted of two steps. Five targets were randomly placed on the map. During the first step, the participant located each target, as in the *locate* task. During the second step, the participant located each target a second time.

These tasks covered traditional target acquisition (*select*), browsing speed (*count*), spatial understanding (*locate* and *relate*), and spatial memory (*relocate*).

Participants performed the study tasks using the following techniques*: edge projection, neighborhood browsing, touch-and-speak,* and *VoiceOver*. To reduce learning time, *bimanual context menus* were excluded from edge projection and the *nearby* command was excluded from touch-and-speak. We adapted Apple's VoiceOver technique to the Microsoft Surface. VoiceOver enables users to select targets by touching them directly, or by moving an on-screen cursor through a list of targets using swipe gestures.

Each participant performed five trials of each task for each technique. *Locate*, *count*, *relate*, and *select* tasks were presented in random order within each technique block. The *relocate* task was presented last within each technique block. Map locations were randomized for each *locate*, *count*, *relate*, and *select* trial, and were randomized once per each technique block for the *relocate* task. Trials began when the participant first touched the screen, and ended when the participant tapped a *Finish Task* button in the lower left corner of the screen.

Following each technique block, participants rated the technique using Likert-type scales (described below). After all techniques had been tested, participants ranked the techniques in order of preference and provided general feedback. Questionnaires were administered verbally, and the experimenter recorded the answers in a text file. The experiment took between 1.5 and 2 hours to complete.

Results

We present performance results for each of the techniques, participants' ratings for each of the techniques, and our observations of common interaction styles and challenges.

Data Collection

Each participant performed 5 trials of the 5 tasks for each of the 4 techniques. Because the *relocate* task featured two steps, two trials were recorded for that task. One participant was unable to test VoiceOver due to time constraints. A total of 1650 trials were recorded. For each trial, we recorded the start time, end time, all touch events, and participant responses for the *count*, *relate*, and *select* tasks.

Completion Time

Because trial completion time was not normally distributed (Shapiro-Wilk W=0.75, p<.01), we used the nonparametric Aligned Rank Transform (ART) [13,24,33] on completion time. The ART enables parametric F-tests to be used on nonparametric data, while preserving the correctness of interaction effects. After using the ART, the resulting data was analyzed using a mixed-effects REML model, which preserves large denominator degrees-of-freedom but compensates with wider confidence intervals. Pairwise comparisons used Holm-Bonferroni correction [14].

We found significant effects of *technique* ($F_{3,1591}$=43.47, p<.0001) and *task* ($F_{5,1589}$=133.95, p<.0001) on completion time, as well as an interaction between *technique* and *task* ($F_{15,1589}$=15.55, p<.0001). The average completion time for each technique is shown in Figure 6. Pairwise comparisons revealed that touch-and-speak was faster than VoiceOver and neighborhood browsing ($F_{1,1592.66}$=26.05, p<.01; $F_{1,1589.05}$=91.32, p<.01) and that edge projection was faster than VoiceOver and neighborhood browsing ($F_{1,1592.66}$=27.96, p<.01; $F_{1,1589.05}$=94.96, p<.01).

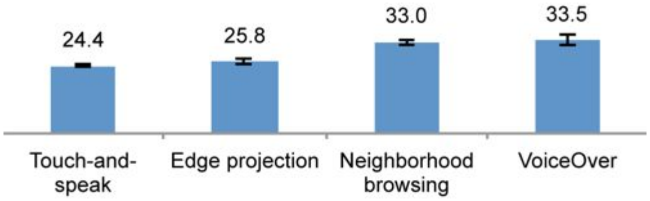


Figure 6. Task completion time across all tasks in seconds. Lower is better. Error bars show ±1 SE.

We also computed pairwise differences for each task:

- Touch-and-speak was faster than VoiceOver for the *locate, count,* and *select* tasks ($F_{1,267}$=21.07, p<.05; $F_{1,267}$=10.04, p<.05; $F_{1,267}$=8.05, p<.05).
- Touch-and-speak was faster than neighborhood browsing for the *count* and *select* tasks ($F_{1,267}$=145.95, p<.05; $F_{1,267}$=8.67, p<.01).
- Touch-and-speak was faster than edge projection for the *count* task ($F_{1,267}$=10.08, p<.01).
- Edge projection was faster than VoiceOver for the *locate* and *relate* tasks ($F_{1,267}$=12.75, p<.01; $F_{1,267}$=8.77, p<.01).
- Edge projection was faster than neighborhood browsing for the *count* task ($F_{1,267}$=79.32, p<.01).
- Neighborhood browsing was faster than VoiceOver for the *locate* task ($F_{1,267}$=11.38, p<.01).
- VoiceOver was faster than neighborhood browsing for the *count* task ($F_{1,267}$=75.37, p<.01).

There was a marginal time difference between the two steps of the *relocate* task ($F_{1,510}$=3.68, p=0.056). For the first step, touch-and-speak, edge projection, and neighborhood browsing were all significantly faster than VoiceOver ($F_{1,267}$=8.94, p<.01; $F_{1,267}$=8.10, p<.01; $F_{1,267}$=10.10, p<.01). On the second step, there was no significant effect of *technique* on task time ($F_{3,267}$=1.01, p=.38, *n.s.*). Note that while VoiceOver was significantly slower than the other techniques at first, it approached the speed of the other techniques for the second step. Figure 7 shows the average task completion time for the *relocate* task.

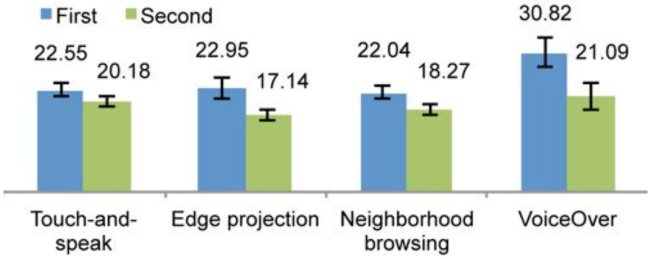


Figure 7. Task completion times in seconds for the two steps of *relocate*. Lower is better. Error bars show ±1 SE.

Correct Answer

Each trial was marked as correct or incorrect. For *locate* and *relocate* tasks, a trial was correct if the participant touched the specified target on the screen. For other tasks, a trial was correct if the participant provided the correct response to the question. As with completion time, we performed an ART procedure and analyzed the results using a mixed-effects REML model. There was a significant difference in correct answers by *technique* ($F_{3,1596}$=131.34, p<.0001) and *task* ($F_{5,1589}$=77.70, p<.0001), as well as a significant interaction between *technique* and *task* ($F_{15,1589}$=23.11, p<.0001). Pairwise comparison showed that there were significantly more incorrect answers when using VoiceOver than when using touch-and-speak, edge projection, or neighborhood browsing ($F_{1,1601.33}$=7.31, p<.05; $F_{1,1601.33}$=235.20, p<.05; $F_{1,1601.33}$=27.70, p<.05). Figure 8 shows the overall correct percentage by technique.

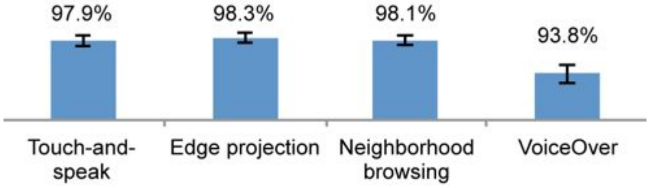


Figure 8. Correct answer percentage by technique. Higher is better. Error bars show ± 1 SE.

Subjective Ratings

Participants rated each technique by indicating their agreement with a series of statements using a 7-point Likert-type scale (*1=strongly disagree, 7=strongly agree*). The following statements were used: this technique was *enjoyable*; this technique was *fast*; this technique was *accurate*; this technique was *frustrating*; this technique was *useful*; this technique was *easy to understand*; this technique *helped me to understand the map*.

Likert ratings were analyzed using a Friedman nonparametric test, with Wilcoxon signed-rank tests for pairwise comparisons. Because one participant did not complete all four techniques, that participant's data was excluded from this analysis. Significant results were found for the following measures: *enjoyable, fast, useful,* and *easy to understand* ($\chi^2_{3,N=13}$=15.27, p<.01; $\chi^2_{3,N=13}$=13.53, p<.01; $\chi^2_{3,N=13}$=8.74, p<.05; $\chi^2_{3,N=13}$=8.31, p<.05). Pairwise comparison found the following significant differences: touch-and speak was significantly more enjoyable than VoiceOver (Z=2.85, N=13, p<.05), and touch-and-speak was perceived as faster than VoiceOver (Z=2.85, N=13, p<.05). No other pairwise differences were found.

Participants also ranked each of the techniques that they tested from most to least favorite. A Friedman test revealed a significant effect of *technique* on rank ($\chi^2_{3,N=13}$=15.92, p<.001). Pairwise comparison showed that touch-and-speak and edge projection were ranked significantly better than VoiceOver (Z=3.00, N=13, p<.05; Z=3.25, N=13, p<.05). Figure 9 shows the number of first and second choice picks per each technique.

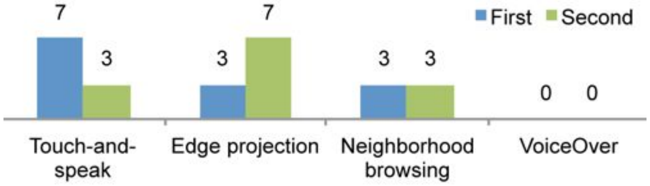


Figure 9. Number of participants who chose each technique as their first or second preferred choice.

General Observations and Participant Feedback

In general, participants varied widely in their search strategies. However, we noticed a number of challenges that were commonly experienced when using the touch screen interfaces. Regardless of the technique used, searching the entire screen to find an item was difficult, time-consuming, and often appeared to frustrate participants. Interestingly, many participants seemed to adopt and adhere to suboptimal search strategies. Most participants explored the screen in an unsystematic, random fashion, causing them to revisit places that they had already explored. More than half of the participants explored the screen by tapping repeatedly, rather than dragging their finger over the screen, which sometimes caused them to pass over and miss a target while their finger was raised. These strategies may have been adopted due to participants' lack of familiarity with touch screens, or may have matched their strategies for searching a physical space (*i.e.,* by "feeling around").

Participants also varied considerably in the speed with which they moved their hands. In some cases, moving too quickly could be confusing, such as when users moved their hands quickly across the screen and through a target, but had moved past it by the time its name had been read. Users then sometimes struggled to locate the target again.

Although participants were told that they could use one or two hands, most used only one hand at a time. However, some participants used their non-dominant hand to tap the *Finish Task* button, while others used their non-dominant hand to "measure" the distance between targets during the *relate* task. In other cases, participants rested their non-dominant hand at the edge of the screen, but sometimes were confused when they accidentally touched the screen and performed some unintended action.

Edge projection overlay. Participants often commented positively about this technique while using it. Most participants relied primarily on the bottom edge closest to themselves, which seems reasonable, as it required them to stretch the least, although one left-handed participant primarily used the left edge. Participants sometimes "missed" an item when browsing along the edge because they started their search from an arbitrary point along the edge rather than from the corner. One reason for this behavior may have been to avoid accidentally activating the *Finish Task* button, although participants were told that nothing "bad" would happen if they did accidentally press the button. Very few participants used two hands to triangulate on-screen targets. Furthermore, although it was possible to use the position of the edge proxies to quickly find the leftmost or topmost target in the *relate* task, most participants did not discover this advanced strategy, and instead located each target separately on the screen.

Neighborhood browsing overlay. Although neighborhood browsing was popular with participants (three chose it as their favorite technique, and another three chose it as their second favorite), participants often were slow in using this technique. Neighborhood browsing was especially slow for *count* tasks, as the participant needed to explore the entire screen to count all of the items. One possible explanation is that participants had difficulty creating the proper mental model for the screen layout. Also, participants often used more than one finger or their whole hand to search the screen (consistent with prior research [34]), and in so doing accidentally activated the guided directions feature. Despite these difficulties, the guided directions were extremely popular, and participants were overall quite effective in following them. Many participants commented positively about this feature; for example, one participant stated, "I just love the coaching to find things on screen."

Touch-and-speak overlay. This technique was popular, and participants could use this technique quickly. As with neighborhood browsing, participants enjoyed following the guided directions. Some participants unsuccessfully tried to invent additional commands, such as saying "list hotels" to list only the hotels on the map, or performed natural language queries such as, "What time does Store 'X' open?" However, these problems were quickly resolved. Many participants noted that they liked this technique, but would not use it in all situations, such as in noisy environments, or in public places (due to privacy concerns).

VoiceOver. Participants experienced several problems when using VoiceOver. Participants found the two selection modes (swiping to move through the list of targets, and directly touching a target to select it) to be particularly confusing, and were often unsure which of the two actions they had taken. Often participants would perform a swipe gesture, hear the name of a target that they had selected, and then be puzzled about how to actually locate that target on the screen. Swipe gestures were also performed with considerable variation, and thus were sometimes misrecognized. Although the list of targets was ordered from left to right, and thus participants could swipe through the list to quickly answer *relate* questions, few did so.

In general, because VoiceOver provided no additional support for locating targets, tasks that required participants to locate on-screen targets (*e.g., locate* and *relocate*) were particularly challenging. Participants described this as "a pain," and one participant was forced to give up on a *locate* task after being unable to find the target.

DISCUSSION

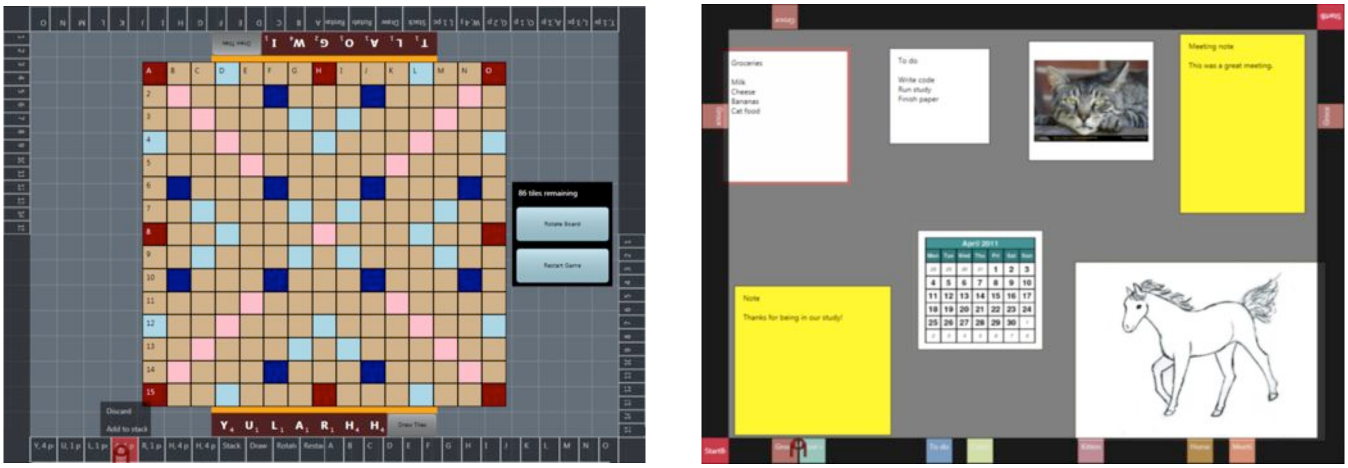
As mentioned previously, using touch screen interfaces remains a significant and sometimes intimidating challenge for blind people. At the start of the study session, numerous participants mentioned their lack of skill with touch screens, and one participant stated that she was "terrified of touch screens." Improving the usability, accessibility, and approachability of touch screens could therefore significantly help this currently excluded population.

Our current research began with our assertion that touch screen interfaces *can* be made more effective for blind people. We introduced three new accessible touch screen interaction techniques that improve upon current techniques in several ways: two of the three techniques performed significantly faster than a popular commercial technique, all three techniques resulted in greater spatial understanding of the screen layout, and all three techniques were preferred to the commercial technique for interacting with a large touch screen application.

In spite of this general success, it is difficult to declare a clear winner from our current evaluation. While touch-and-speak was preferred by participants and performed faster overall, many participants stated that they would not be comfortable using that technique in some contexts. In addition, edge projection was marginally faster for the *relate* task. It is possible that there is no "best" solution, and that different access overlays may be appropriate for specific scenarios, much as PC-based screen readers have multiple modes. When asked to choose their preferred technique, multiple participants said that they would most prefer the ability to combine or to switch among techniques.

Design Recommendations

We designed access overlays to support the criteria of *preserving spatial layout*, *leveraging bimanual interaction*, and *reducing search space* in *walk-up-and-use scenarios*. Our study results reaffirm the importance of these criteria for developing usable touch screen interfaces for blind



Groceries

Milk
Cheese
Bananas
Cat food

To do

Write code
Run study
Finish paper

Meeting note

This was a great meeting.

Note

Thanks for being in our study!

Figure 10. Board game (left) and bulletin board (right) applications, made accessible using an edge projection overlay.

people. In addition to supporting these criteria, our study results and observations suggest additional design guidelines for accessible touch screens:

- *Allow users to quickly switch between linearized and two-dimensional navigation.* Neighborhood browsing was especially slow for tasks such as *count* because it provided no easy way to scan through on-screen targets.
- *Avoid implicit mode switching.* Switching between modes should be clear and deliberate. In VoiceOver, which used both direct touch and swipe gestures, participants sometimes accidentally performed one or the other, and were confused by the result.
- *Avoid distinctions based on the number of fingers used.* Participants often used multiple fingers to explore the screen or inadvertently brushed the side of their hand against the screen. Multi-finger gestures such as split-tap should not be used to change modes, or should require some type of confirmation to avoid accidental activation.

New Accessible Touch Screen Applications

As access overlays offer new opportunities for blind people to interact spatially with touch screens, we are developing a number of applications to demonstrate these techniques. We describe three such applications, two of which are illustrated in Figure 10.

Map. We developed an interactive map that allows users to explore cities, countries, and points of interest using each of the access overlays. In the city view, users may request walking directions to local points of interest and send these directions to their phone via SMS. This prototype was recently demonstrated at the 2011 *CSUN International Technology and Persons with Disabilities Conference,* where it was tested by approximately a dozen blind people.

Board game. We have added access overlays to a prototype board game application, specifically the Scrabble crossword game. Currently, blind game players are often restricted to specialized Braille versions of board games, and many of our study participants were excited about the possibility of using a touch screen to play games.

Bulletin board. The bulletin board application allows users to explore notes, calendars, and images on a virtual bulletin board. This interface, built using a modified version of the Microsoft Surface's scatter view control, could be used in the future to organize and share other types of documents.

FUTURE WORK

The results of our study suggest several opportunities to refine our existing access overlays. Neighborhood browsing might benefit from a more robust layout algorithm (*e.g.,* Starburst [2]). Touch-and-speak might benefit from additional commands, such as searching within a user-defined region or searching for items of a specific type.

In addition to extending existing overlays, the system of access overlays could also be extended to utilize additional feedback channels, such as multiple synthesized voices, individualized audio feeds [11,21], or physical add-ons such as SLAP widgets [31]. Combining well-designed audio feedback with additional feedback modalities could significantly improve overall performance.

Current access overlays support limited bimanual input, but do not provide appropriate output when the user is touching the screen with both hands. Some study participants used both hands to search the screen, but were unsure which hand (or part of the hand) touched an item. Following such an encounter, one participant stated, "I touched it, but I don't know where I touched it." This problem may be resolved by providing richer feedback about where items are on the screen when they are touched.

Currently, access overlays assume that on-screen targets remain in place while the user explores the screen. The user is not notified when on-screen items move. Future versions could enable users to track the movement of on-screen objects over time, enabling access to dynamic content.

While large touch screens are often used collaboratively by multiple users, access overlays are currently designed for a single user. In the future, access overlays may be extended to support collaborative use, either between multiple blind users or between blind and sighted collaborators.

Finally, access overlays may be extended to general applications on the Microsoft Surface or other touch screen devices. Although the applications presented in this study were specially constructed, future versions could automatically capture on-screen targets using existing accessibility APIs, and thus could provide access to many standard touch applications.

CONCLUSION

While current touch screen accessibility techniques provide basic access to touch screens, we suggest that an ideal touch screen interface must also address issues such as spatial understanding, especially on large touch screen displays. We have introduced three techniques, called access overlays, which are optimized for exploring spatial interfaces on large touch screens. Our evaluation showed that access overlays enabled users to locate on-screen targets faster than traditional techniques, improved spatial understanding, and were preferred. This work introduces new approaches for improving touch screen accessibility.

REFERENCES

1. Bau, O., Poupyrev, I., Israr, A., and Harrison, C. TeslaTouch: electrovibration for touch surfaces. *UIST '10*, 283-292.
2. Baudisch, P., Zotov, A., Cutrell, E., and Hinckley, K. Starburst: a target expansion algorithm for non-uniform target distributions. *AVI ’08*, 129-137.
3. Bonner, M., Brudvik, J., Abowd, G., and Edwards, W.K. No-Look Notes: accessible eyes-free multi-touch text entry. *Pervasive ’10*, 409-427.
4. Buxton, W., Foulds, R., Rosen, M., Scadden, L., and Shein, F. Human interface design and the handicapped user. *SIGCHI Bulletin 17*, 4 (1986), 291-297.
5. Buxton, W., Hill, R., and Rowley, P. Issues and techniques in touch-sensitive tablet input. *SIGGRAPH ’85*, 215-224.
6. Calder, M., Cohen, R.F., Lanzoni, J., and Xu, Y. PLUMB: an interface for users who are blind to display, create, and modify graphs. *ASSETS ’06*, 263-264.
7. Carew, S. Touch-screen gadgets alienate blind. *Reuters* (2009). http://reut.rs/gHji5X
8. Fortune, S. A sweepline algorithm for Voronoi diagrams. *Algorithmica 2*, 1 (1987), 153-174.
9. Grossman, T. and Balakrishnan, R. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor’s activation area. *CHI ’05*, 281-290.
10. Guerreiro, T., Lagoa, P., Nicolau, H., Gonalves, D., and Jorge, J. From tapping to touching: making touch screens accessible to blind users. *IEEE Multimedia 15*, 4 (2008), 48-50.
11. Hancock, M.S., Shen, C., Forlines, C., and Ryall, K. Exploring non-speech auditory feedback at an interactive multi-user tabletop. *GI ’05*, 41-50.
12. Hart, P.E., Nilsson, N.J., and Raphael, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics 4*, 2 (1968), 100-107.
13. Higgins, J.J. and Tashtoush, S. An aligned rank transform test for interaction. *Nonlinear World 1*, 2 (1994), 201-211.
14. Holm, S. A simple sequentially rejective Bonferroni test procedure. *Scandinavian Journal of Statistics 6*, (1979), 65-70.
15. Jagdish, D., Sawhney, R., Gupta, M., and Nangia, S. Sonic Grid: an auditory interface for the visually impaired to navigate GUI-based environments. *IUI ’08*, 337-340.
16. Kane, S.K., Bigham, J.P., and Wobbrock, J.O. Slide Rule: making mobile touch screens accessible to blind people using multi-touch interaction techniques. *ASSETS ’08*, 73-80.
17. Kane, S.K., Jayant, C., Wobbrock, J.O., and Ladner, R.E. Freedom to roam: a study of mobile device adoption and accessibility for people with visual and motor disabilities. *ASSETS ’09*, 115-122.
18. Kane, S.K., Wobbrock, J.O., and Ladner, R.E. Usable gestures for blind people: understanding preference and performance. *CHI ’11*, 413-422.
19. Landau, S. and Wells, L. Merging tactile sensory input and audio data by means of the Talking Tactile Tablet. *Eurohaptics ’03*, 414-418.
20. Loomis, J.M., Golledge, R.G., and Klatzky, R.L. Navigation system for the blind: auditory display modes and guidance. *Presence 7*, 2 (1998), 193-203.
21. Morris, M.R., Morris, D., and Winograd, T. Individual audio channels with single display groupware: effects on communication and task strategy. *CSCW ’04*, 242-251.
22. Plimmer, B., Crossan, A., Brewster, S.A., and Blagojevic, R. Multimodal collaborative handwriting training for visually-impaired people. *CHI ’08*, 393-402.
23. Reisinger, D. Universities reject Kindle over inaccessibility for the blind. *CNET* (2009). http://cnet.co/gSjyv9
24. Salter, K.C. and Fawcett, R.F. The ART test of interaction: a robust and powerful rank test of interaction in factorial models. *Communications in Statistics-Simulation and Computation 22*, 1 (1993), 137-153.
25. Su, J., Rosenzweig, A., Goel, A., de Lara, E., and Truong, K.N. Timbremap: enabling the visually-impaired to use maps on touch-enabled devices. *MobileHCI ’10*, 17-26.
26. Talbot, M. and Cowan, W. On the audio representation of distance for blind users. *CHI ’09*, 1839-1848.
27. Tse, E., Hancock, M., and Greenberg, S. Speech-filtered bubble ray: improving target acquisition on display walls. *ICMI ’07*, 307-314.
28. U.S. Bureau of Engraving and Printing. EyeNote. (2011). http://moneyfactory.gov/images/EyeNote_Press_Release_4_4-19_2_4.pdf
29. Vanderheiden, G.C. Use of audio-haptic interface techniques to allow nonvisual access to touchscreen appliances. *HFES 40*, (1996), 1266.
30. Wall, S. and Brewster, S. Feeling what you hear: tactile feedback for navigation of audio graphs. *CHI ’06*, 1123-1132.
31. Weiss, M., Wagner, J., Jansen, Y., Jennings, R., Khoshabeh, R., Hollan, J.D., and Borchers, J. SLAP widgets: bridging the gap between virtual and physical controls on tabletops. *CHI ’09*, 481-490.
32. Wigdor, D., Williams, S., Cronin, M., Levy, R., White, K., Mazeev, M., and Benko, H. Ripples: utilizing per-contact visualizations to improve user interaction with touch displays. *UIST ’09*, 3-12.
33. Wobbrock, J.O., Findlater, L., Gergle, D., and Higgins, J.J. The Aligned Rank Transform for nonparametric factorial analyses using only ANOVA procedures. *CHI ’11*, 143-146.
34. Wobbrock, J.O., Morris, M.R., and Wilson, A.D. User-defined gestures for surface computing. *CHI ’09*, 1083-1092.
35. Worden, A., Walker, N., Bharat, K., and Hudson, S. Making computers easier for older adults to use: area cursors and sticky icons. *CHI ’97*, 266-271.
36. Yfantidis, G. and Evreinov, G. Adaptive blind interaction technique for touchscreens. *Universal Access in the Information Society 4*, 4 (2006), 328-337.