Tactile Graphics with a Voice: Using QR Codes to Access Text in Tactile Graphics

Catherine M. Baker, Lauren R. Milne, Jeffrey Scofield, Cynthia L. Bennett, Richard E. Ladner
Computer Science & Engineering
University of Washington Box 352350
Seattle, WA 98195-2350
{cmbaker, milnel2, jeffsco, bennec3, ladner}@cs.washington.edu

ABSTRACT

Textbook figures are often converted into a tactile format for access by blind students. These figures are not truly accessible unless the text within the figures is also made accessible. A common solution to access text in a tactile image is to use embossed Braille. We have developed an alternative to Braille that uses QR codes for students who want tactile graphics, but prefer the text in figures be spoken, rather than in Braille. Tactile Graphics with a Voice (TGV) allows text within tactile graphics to be accessible by using a talking QR code reader app on a smartphone. To evaluate TGV, we performed a longitudinal study where ten blind and low vision participants were asked to complete tasks using three alternative picture taking guidance techniques: 1) no guidance, 2) verbal guidance, and 3) finger pointing guidance. Our results show that TGV is an effective way to access text in tactile graphics, especially for those blind users who are not fluent in Braille. In addition, guidance preferences varied with each of the guidance techniques being preferred by at least one participant.

Categories and Subject Descriptors
H.5.2. User Interfaces.

General Terms
Design, Human Factors

Keywords
Access technology; blind; camera; non-visual feedback; visually impaired; tactile graphics; QR codes.

1. INTRODUCTION

From pictures of plant cells to diagrams of parabolas, images are an integral part of most textbooks, and frequently convey information that cannot be understood from text alone. Therefore, these images and the text contained within them should be accessible to all students, thus there is a need to create alternative access methods for people with disabilities. The common solution when making a textbook accessible for blind students is to create tactile representations of the images, or tactile graphics. Tactile graphics can be low fidelity by using spaghetti glued on poster board to a high fidelity graphic printed on an embossing printer. Studies have shown that tactile graphics are valuable for conveying graphical information [8]. In a survey of 24 teachers that worked with visually impaired children, all indicated that there were situations where tactile graphics were important for and effective at teaching a lesson [20]. Teachers also indicated that the ability to explore graphics, discover the information, and answer questions about the information independently was a fundamental part of the learning process [19, 20].

ASSETS '14, October 20 - 22 2014, Rochester, NY, USA
Copyright 2014 ACM 978-1-4503-2720-6/14/10…$15.00
http://dx.doi.org/10.1145/2661334.2661366

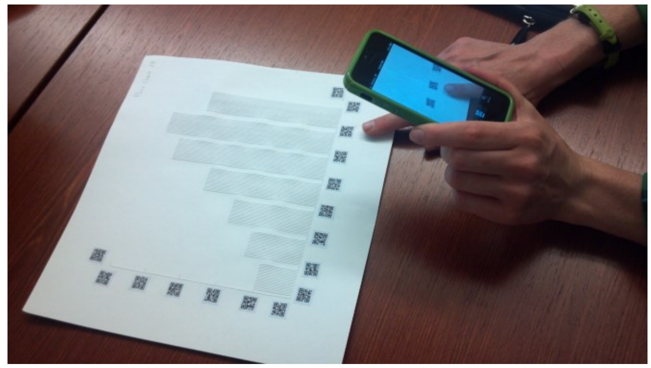

Figure 1. The Tactile Graphics with a Voice system in use. The subject is using the finger pointing mode to select which QR code to scan.

The text in tactile graphics is typically represented using embossed Braille. However, a 2009 report by the National Federation of the Blind states that less than forty percent of the functionally blind population in the United States is fluent in Braille [14]. Therefore tactile graphics with Braille labels are not accessible to a significant number of blind people.

There have been a few solutions to this problem presented by the access technology community. Examples include a system where an overlaid tactile graphic on a tablet gives audio feedback when touched [11] and a talking pen to explore a tactile graphic [12]. However, these solutions require using specialized devices, which can be expensive.

We present a new system for embedding and accessing text in tactile graphics using QR codes, which are small codes that directly encode textual information (Figure 1). QR codes can be read by a smartphone and can easily be created by anyone with access to a computer. We created a smartphone application for blind users called Tactile Graphics with a Voice (TGV) that scans

QR codes and provides feedback to help users aim the smartphone camera. We conducted interviews and surveys with people who are blind or low vision to design the application and determine what types of non-visual feedback are most helpful to aim the smartphone. In addition, we developed a finger pointing method to help determine which QR code should be read when there are multiple QR codes in the camera view.

We evaluated our application in a longitudinal study and found that people who are blind or low vision were able to successfully answer questions about tactile graphics by scanning QR codes. Key findings from the study are listed below.

1. Four of our participants were able to correctly answer questions about the images using the QR codes, but were not able to use the Braille equivalents as they were not fluent in Braille.
2. Participants fluent in Braille spent an equivalent amount of time on tasks and had similar accuracy for both the QR codes and Braille equivalents.
3. Preferences varied greatly among participants as to what kind of feedback from the smartphone application is most helpful. Four of our participants preferred the Silent mode, four preferred the Finger Pointing mode and two preferred the Verbal mode.

Our contributions are:

1. The development of Tactile Graphics with a Voice (TGV), a system to access text on tactile graphics using QR codes and a smartphone application.
2. The findings from our study, which show that blind and low vision users support having a variety of non-visual feedback mechanisms to help aim a camera.

2. RELATED WORK

We discuss prior work related to three areas of our system: (i) methods to embed textual information on tactile graphics, (ii) methods to access the information and (iii) use of the finger pointing technique as a means to select which information to be read aloud.

2.1 Accessing Textual Information on Tactile Graphics

Braille labels on educational tactile graphics present difficulties for both students and teachers. Sheppard and Aldrich [1,20] found that both students and teachers had difficulties with Braille labels on tactile graphics in the classroom. Teachers, in particular, had issues placing the labels without text overlapping the figure. Students struggled with the meaning of a label when it stretched across the entire graphic.

Despite the issues with using only Braille for accessing text, there is little work in the HCI literature using alternative methods. There has been some progress made in the access technology community. Touch Graphics has developed the Talking Tactile Tablet (TTT) [11], a touch-sensitive table on which a user can place a tactile graphic and hear audio information upon touch.. However, this method requires a large touch sensitive surface (~12×15 inches, 6.5 pounds) and has to be connected to a computer via USB which contains the information for the tactile graphic to be explored. Touch Graphics also created the Talking Tactile Pen (TTP)[12], which allows blind users to access information on custom tactile graphics tagged with a proprietary code. The pen contains a small camera used to photograph the proprietary codes. When the pen contacts a tagged area, it reads aloud the corresponding file stored on the pen. Despite the pen's portability in comparison to the TTT, it is a specialized device, and is only useful on properly tagged tactile graphics that have their information stored on the pen. TGV is a solution that attempts to solve the same problem by using non-proprietary codes and a non-specialized, portable, mainstream device like the smartphone.

Voiceye codes are also being used to encode text on graphics[7]. While not used on tactile graphics, they are used in South Korea to make government forms accessible. These are similar to QR codes, but may contain more information for a given area. Users scan these codes with a smartphone application and the corresponding text appears for reading aloud or visual magnification. However, users are not given feedback to assist in scanning the code and the codes must be created with expensive proprietary software. TGV provides a major benefit over current approaches because QR codes can be freely created.

2.2 Camera Use by Blind People

TGV requires the use of a smartphone camera, because it enables the use of QR codes. While aiming the smartphone is a challenging task for blind people, there are research efforts in the accessibility community to tackle this problem. Bigham *et al.* created an application called VizWiz::LocateIt [2], which allows blind users to locate objects using the camera on their smartphone. VizWiz::LocateIt uses crowdsourcing to identify the object in the photo and computer vision techniques to provide audio feedback about the proximity to the object. Our application uses similar audio feedback to guide users to the QR code, but does not rely on crowdsourcing, thus providing quicker feedback.

Using computer vision techniques exclusively with a smartphone camera may enhance camera feedback. Jayant *et al.* created EasySnap [9], a camera application that assists users in taking pictures by providing audio feedback. EasySnap uses computer vision to locate people or objects in the viewfinder and relays information their location and their size in proportion to the viewfinder. They followed with another application, PortraitFramer, which incorporates features of EasySnap and uses haptic feedback communicate where in the viewfinder the people or objects are located. They found that it took little training for users to take better pictures. A similar feature has been built into the camera application on recent versions of iOS [6]. When text-to-speech is enabled, the camera application provides feedback about faces, such as "face at top of screen," to guide users in taking portraits. Our application also incorporates audio feedback, but because users are using their sense of touch to explore a tactile graphic, we decided not to use haptic feedback to avoid cognitive overload.

TGV utilizes audio feedback, but there are diverse options, such as tone and speech. Vasquez *et al.* [22] were interested in learning what type of audio feedback was preferred among blind people using a camera. They considered speech, tone, and no feedback. People strongly preferred speech feedback and found it easier to use than either silent or tone feedback. As a result, we use speech feedback in TGV as opposed to tone.

The majority of the camera applications mentioned above were focused on taking a quality picture of a person or a physical object. Another related space is in technology that allow blind users to scan barcodes, which are similar to QR codes. The majority of commercial applications, such as the i.d. mate[1] or Digit-Eyes,[2] do not provide feedback. However, Tekin *et al.* [21] experimented with different feedback modalities to help blind users scan barcodes on products. They used both verbal feedback and sonification, but their application was evaluated by a single user. TGV distinguishes itself in two ways: 1) our QR codes are labels that can be located by touch and 2) multiple QR codes can be close together.

2.3 Finger Pointing

Because textbook images may have multiple text labels in close proximity of one another, the use of a finger may help select the preferred QR code when multiple codes are in the viewfinder. Thus, we present related work on the practicality of finger pointing as a method to select a preferred QR code.

There are numerous projects that use finger pointing to identify an object or information of interest. One example is the EyeRing [13], a camera worn on the finger that reads information aloud based on where the finger is pointing. Similarly, OrCam[3], also uses finger pointing for people who are low vision. The OrCam is a wearable camera that uses computer vision to identify objects in which a user is pointing and reads aloud information about that object. The manufacturers envision that OrCam can recognize faces, places, objects, and text.

Kane *et al.* developed Access Lens [10], a way for people who are blind or low vision to access documents. This system uses a camera connected to a computer to read aloud the text on documents. Users can point to any element on the document to hear the associated information. This system brings promise to the accessibility of printed documents and demonstrates that finger pointing is an easy way for blind users to control what information they hear. However, Access Lens is not portable. In TGV, we capitalize on finger pointing as a simple means of selecting the information the user wants to hear.

3. FORMATIVE STUDIES

In order to determine the feasibility of substituting QR codes for text labels on tactile graphics, we conducted a survey and follow-up interviews with people who are blind or low vision. We were motivated to learn about the current use of tactile graphics and cameras, and whether people would take interest in using QR codes as labels on tactile graphics.

The online survey was distributed to blind and low vision mailing lists and inquired about their use of Braille, tactile graphics, and camera applications on the smartphone. Twenty-two people completed our survey, where fifteen of our respondents were blind and seven were low vision. There were 12 female and 10 males with an average age of 38.18 (SD=13.46). All of our respondents had taken some college courses, and nine respondents had a graduate degree. Sixteen respondents knew Grade 2 Braille, while only 3 respondents had little to no knowledge of Braille. All but one of the respondents owned a smartphone.

We conducted follow-up interviews with ten of the survey respondents, 6 of them female. We selected a diverse subset of those who indicated they would be willing to be interviewed on the survey. The interviews provided more detail about their survey responses and provided feedback about our proposed system, TGV. The participants' ages ranged from 21 to 67 with an average of 37.6 (SD=13.95). Five participants identified as blind and five as low vision. Five used Braille at work, three knew Braille but did not use it often, and two participants had little familiarity with Braille. Eight participants used tactile graphics in their education and work.

We found that many of our respondents frequently used cameras, especially on smartphones, and were interested in using tactile graphics with QR code labels. Seven of the ten participants reacted positively to replacing Braille with QR codes. One participant noted: "You can fit a lot more information on a QR code than on a Braille label," a sentiment shared by five of our participants.

In addition, many of our survey respondents were familiar with using the camera on their smartphone, and thus have completed similar tasks to scanning QR codes. Fifteen respondents used an application that required the camera on a daily to weekly basis.

We learned that people found non-visual feedback for aiming the camera to be helpful. Just over half the respondents used an application that gave them feedback to help aim the camera. The majority of those respondents indicated that the feedback was helpful, with only one respondent mentioning that he had received feedback that was not helpful as it was unclear what it meant.

In our follow-up interviews, we investigated preferences for feedback modalities on a smartphone camera application: verbal, tonal, haptic, and no feedback. While the participants had a variety of preferences, we found that most participants preferred having the option of a quiet mode. Participants wanted a quiet mode because they felt that expert users needed less feedback, and they would not want to disturb others such as during a meeting.

4. TACTILE GRAPHICS WITH A VOICE (TGV)

TGV is composed of tactile graphics with QR code labels and a smartphone application. The application provides multiple non-visual feedback modalities, and allows the user to select which QR code they want to scan.

4.1 Tactile Graphics with QR Codes

The creation of tactile graphics for TGV requires a similar amount of work as traditional tactile graphics. Traditionally, converting a textbook graphic into a tactile graphics is a labor-intensive process. First, the text must be removed from the graphic. In addition, some extra processing may be needed to make the image understandable in a tactile form. Once the text is removed, it needs to be translated into Braille and be placed back on the image in similar location to the original text. Instead of generating Braille, TGV generates a QR code from text using a free online generator. Because the embosser we used to create the tactile graphics cannot print ink, we printed QR codes on a separate sheet of paper and glued them onto the graphic (See Figure 2 for examples). It was not necessary to mark the QR codes with an embossed symbol because the height difference of the QR codes was sufficient to be felt. If you have an embosser capable of both embossing Braille and printing ink, the only difference from the traditional process is that you would place the QR code labels (with accompanying tactile markers) on the graphic in place of the Braille labels.

[1] http://www.envisionamerica.com/products/idmate/

[2] http://www.digit-eyes.com/

[3] http://www.orcam.com

4.2 Smartphone Application

We created an accessible application for iOS that allows a blind or low vision user to scan a QR code easily, even if there are multiple QR codes close together. The smartphone application is built on top of the ZXing software[4] for scanning QR codes. This software identifies QR codes by looking for an area of black and white variation. We added verbal feedback to help users scan QR codes, as well as the option to use finger pointing to allow the selection of a QR code when many are visible in the viewfinder.

Based on our survey and interviews, we integrated feedback for aiming the camera. In addition, we determined that it was important to have a feedback mode and a silent mode. Because the participants' preferences on feedback modalities varied, we used verbal feedback, based on prior work [22] and that most of our interview respondents indicated that verbal feedback was the easiest to learn.

We presented short clear verbal feedback to assist a user in moving the phone. We based the feedback on the screen location of the QR code, based on related work [22]. For instance, if a participant holds a smartphone in a non-traditional way (e.g. sideways), they will still hear relevant feedback because the navigational instructions to a QR code are based on the current phone orientation.

When multiple QR codes are visible, it is necessary to determine which QR code should be scanned. Therefore, we implemented finger pointing as a method to distinguish which label should be scanned. The selected QR code is the one with the shortest distance to the users' finger. To prevent the application from scanning the incorrect QR code, we set a maximum distance in which a finger can choose a QR code to scan. Unlike Kane, *et al.* [10], which selects the information at the tip of the finger, our application selects the QR code that is closest to any part of the finger. As a result, users needed to be aware of their hand placement to ensure a false positive does not occur.

We identify the finger with color based skin detection [4,16,17]. Because of the constrained black and white environment of tactile graphics, we can identify a painted fingernail by looking for colored pixels and group them as part of the finger.

4.3 Feedback Modalities

Because feedback and finger pointing are not appropriate in every situation, we created three modes for the application: Silent, Verbal, and Finger Pointing.

4.3.1 Silent

Silent mode gives no feedback to help aim the camera. If multiple QR codes are visible in the viewfinder, the application does not scan. When it has successfully scanned a QR code, it chimes and then reads the scan aloud.

4.3.2 Verbal

Verbal mode provides spoken feedback to help aim the camera. If multiple QR codes are visible in the viewfinder, the application speaks this information and does not scan. When the application has successfully scanned a QR code, it chimes and then reads the scan aloud.

4.3.3 Finger Pointing

Finger Pointing mode provides spoken feedback to help aim the camera. The application needs to detect the finger in order to scan. If the finger is not detected, the application speaks this information and does not scan. If multiple QR codes are visible, the application will scan as long as the finger is detected. When the application has successfully scanned a QR code, it chimes and then reads the scan aloud.

5. LONGITUDINAL STUDY

To evaluate the efficacy of TGV, we conducted a six-session longitudinal study with ten blind and low vision participants. Participants answered questions using TGV with the three modes of feedback (Silent, Verbal and Finger Pointing). In the last session, we had participants who knew Braille complete the same tasks using tactile graphics with Braille labels.

5.1 Participants

We conducted the study with ten participants (four male, six female), with ages ranging from 30 to 54 years, and an average age of 41.9 (SD = 8.1). Five had college degrees, three had some college education, and two had a high school education. Four participants identified as low vision and the remaining six identified as blind. Six participants completed the Braille portion of the study, while four were not Braille literate or were not confident in their Braille skills. Overall, participants did not have much experience with tactile graphics, with five never using them, three rarely using them, one using them once per month, and one using them once per week. Nine participants had smartphones; seven had iPhones and two had Androids. Our smartphone users had used camera applications for varying frequencies: two used them daily, two weekly, two monthly, and three rarely used their smartphone cameras. Finally, eight participants had no experience scanning QR codes with their smartphones and two had some experience.

5.2 Apparatus

The TGV application was ran on an iPod Touch 4th generation and an iPhone 5, each running iOS 6. Each participant used the same device for all six sessions. The tactile graphics were printed on standard 11x11.5 inch Braille paper and embossed with a Tiger embosser[5]. QR codes were printed on standard printer paper and cut and pasted onto the tactile graphics in the appropriate places. Braille labels were embossed directly on the graphic using Nemeth code, the type of Braille usually found in math textbooks. Numbers in Nemeth code and Grade 1 and 2 Braille are similar; in Nemeth code the dots are shifted down a row [15].

5.3 Procedure

We had each participant complete six sessions over a two week period. We wanted participants to interact with the application over time to emulate a real-world situation, such as using the application to complete schoolwork.

During the first session, we collected demographic information from the participants, and taught participants how to use the three modes of the TGV application (Silent, Verbal and Finger Pointing). We explained how each mode worked and provided basic information about using the application, such as the suggested scanning height and where the camera was physically

[4] http://code.google.com/p/zxing/

[5] http://www.viewplus.com/

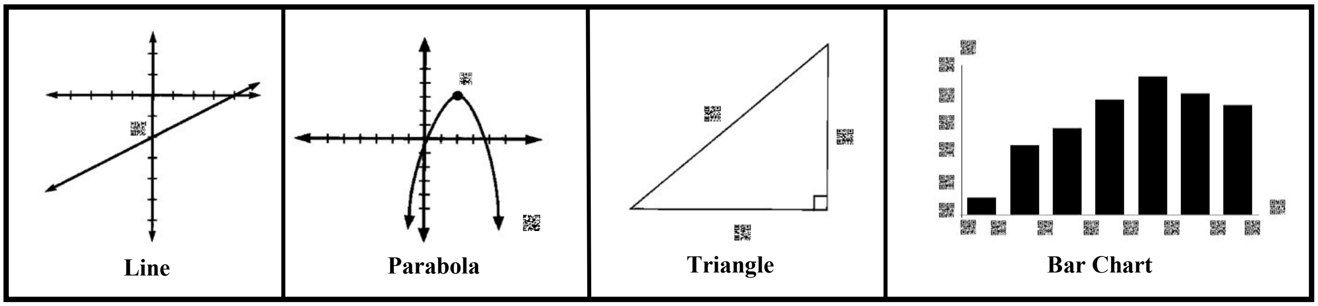

Figure 2. This is an example of each of the tasks that the participants completed. The first task is to find the y-intercept of a line. The second task is to find the (x,y)-coordinates of the vertex of a parabola. The third task is to find the length of the hypotenuse of a right triangle. The fourth task is to find the range of the tallest bar on the bar chart. In each session, participants used similar graphics, but with different labels (i.e. the parabola might be the opposite direction and have a different vertex).

located on the device. Participants had a chance to practice scanning a QR code with each mode.

During each session, participants completed a total of twelve tasks by using each mode of TGV (Silent, Verbal and Finger Pointing) on the following four tasks (Figure 2):

1. *Line.* The first task was to find the y-intercept on a line graph. The graphics always had one QR code representing the value of the intercept.
2. *Parabola.* The next task was to find the (x.y)-coordinates of a parabola vertex. The graphics used in this task always had two QR codes, one for the coordinates of the vertex and one for the equation of the parabola.
3. *Triangle.* The third task was to find the length of the hypotenuse of a right triangle. The graphics in this task always had three QR codes, as the lengths of all sides were labeled.
4. *Bar Chart.* The final task was to find the left and right values on the x-axis of the tallest bar in a bar chart. For this task, the bar chart had seven bars, and there was a QR code marking the bounds of each bar on the x-axis and each tick mark on the y-axis as well as axes labels.

The images for each task were based on images taken from a precalculus textbook [5]. At the beginning of each task, a tactile graphic was placed in front of the participant, and they were instructed to begin. The task ended when the participant responded with their answer. A researcher recorded the task completion time and their answer. We video recorded participants to validate this data. For the first three tasks (Line, Parabola, and Triangle), participants can only receive 0 or 100% accuracy. On the final task (Bar Chart) task, participants can also receive 50% accuracy, as that task required finding both the left and right values of a range. Participants were not told whether or not their answers were correct to mimic a testing situation. We randomized the order of modes used in each session but kept the order of tasks consistent: Line, Parabola, Triangle, and Bar Chart. At the end of each session, we conducted a survey to gauge the participants' preferences for the feedback modes.

In the last session, participants who were proficient in Braille attempted to complete the same tasks using Braille labels in lieu of the QR codes. We choose to have the comparison to Braille only in the last session for two reasons. The first is that participants were already familiar with Braille so we felt that they did not need the time to learn it. By completing the sessions with TGV, they would be familiar with the tasks by the sixth session, making the comparison from TGV to Braille more equal. The second is we wanted to limit the length of the sessions to prevent fatigue. As some of our participants were Braille-literate, but not familiar with Nemeth code, we explained the difference between Nemeth and Braille to those participants.

5.4 Design and Analysis

The study was a 6×3 within-subjects design with factors for *Session* and *Mode*. The levels of *Session* were (1-6); the levels for *Mode* were (Silent, Verbal, Finger Pointing). Each participant completed a total of 72 trials, for a total of 720 trials with the smartphone, and six participants additionally performed 4 trials with Braille at the end of the session. The other four participants were not comfortable enough with Braille to attempt those tasks. We measured completion time and accuracy for each task. If participants took longer than 180 seconds to complete a task, we stopped them and recorded that they had timed-out on the task. Participants were still allowed to submit an answer if they timed-out on the task.

While analyzing completion time for a task, we used a mixed-effects model analysis of variance with fixed effects of *Session* and *Mode*, with *Participant* modeled as a random effect. For accuracy and preference data, we looked at the descriptive statistics.

6. RESULTS

6.1 Accuracy

The accuracy for each task did not vary across the different modes (Silent mode: 88%, Verbal mode: 88%, Finger Pointing mode: 89%).While there was no significant difference in the accuracy between the first and last session, we found that accuracy tended improve in the last session (Figure 3). In addition, we saw that the accuracy tended to be lower on the bar chart task (Table 1). We hypothesize this was the case because this task had the most QR codes closest together.

Table 1. The table shows the average accuracy across all sessions for each mode and each task.

	Silent	Verbal	Finger Pointing
Line	97%	97%	93%
Parabola	93%	95%	95%
Triangle	88%	88%	90%
Bar Chart	73%	70%	79%
All Tasks	88%	88%	89%

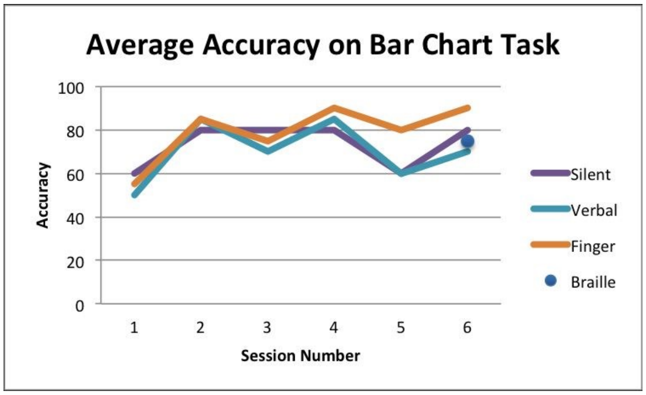

Figure 3. A comparison of the average accuracy for the Bar Chart task across the six sessions for the three modes (n=10) and Braille (n=6) on the last session. Participants were asked to find the range of the tallest bar and their answer could be 0, 50 or 100% correct.

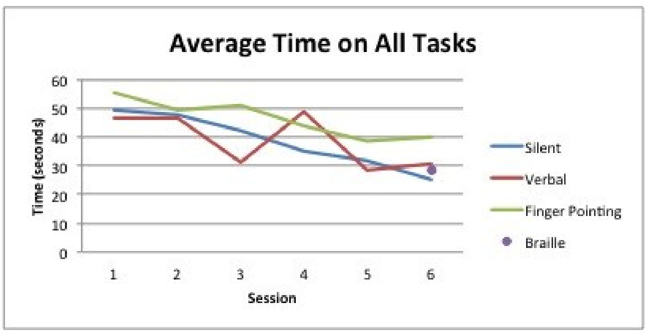

Figure 4. A comparison of the average time it took for each participant to give the answer for a task for the three modes (n=10) across the six sessions as well as for Braille (n=6) on the final session.

6.2 Time

If participants reached 180 seconds without answering the question and completing the task, this was counted as a time-out, and the time is not included in the average or the statistical analysis. Out of 720 tasks, the total number of tasks that timed out was 41, or 5.7%, and almost half of those time-outs (19) occurred in the first session. Additionally, over half of the time-outs (21) occurred during the difficult Bar Chart task. The time-outs occurred in all the modes, with 16 time-outs occurring in the Finger Pointing mode, 16 occurring in the Silent mode, and 9 occurring in the Verbal mode.

With time-outs removed, the average QR code task completion time for all sessions and all modes was 40.9 seconds (SD =36.3). However, the participants were faster in the sixth session than the first (see Figure 4) and the effect of *Session* on time was statistically significant ($F_{5,\ 640}$=2.268, p<.05). In session six, the average Silent mode completion time was 25.3 seconds (SD=18.7), Verbal mode completion time was 30.5 seconds (SD=29.8) and Finger Pointing mode completion time was 40.3 seconds (SD=29.4). The effect of *Mode* on time was not statistically significant ($F_{2,\ 640}$=0.619, p=.5391), though it is observed that Finger Pointing mode took more time than the Verbal and Silent modes.

6.3 Feedback Modality Preference

At the end of each session, we asked each participant to indicate their feedback modality preference by ranking the different modes. In addition, each participant to rated how much they liked using each mode on a 7-point Likert scale, where a rating of 1 meant that the participant liked the mode and a 7 meant they did not like it at all. Like our survey and interview, we found a wide range of preferences.

Despite the wide range of preferences, we found that Verbal mode received an average rating of 2.87 (SD=1.57), Finger Pointing mode received an average rating of 3.63 (SD=1.71) and Silent mode received an average rating of 3.98 (SD=2.11). However, the ranking of each method varied strongly between participants and over time. In the final session of the study, four out of ten participants ranked Silent mode as their favorite mode, four ranked the Finger Pointing mode as their favorite and two ranked the Verbal mode as their favorite.

Participants that preferred the Finger Pointing mode generally thought it was more accurate. Participant 1 stated:

I like the concept of the finger pointing. I feel more confident that since it looks for a finger it's getting the right QR code if you have multiple on the same page.

People who did not like the Finger Pointing mode thought it was difficult to use. Participant 3 stated that "I haven't been able to see my finger point in years, so knowing where my finger is isn't useful," but thought that it might be useful for others:

I did like that you're - that it's trying to branch out and give people options for identifying things like with a finger. It's a pretty neat touch. I like that. I could see that turning into something useful. I think my preference was still for just taking it with a simple picture with the camera

Participants that preferred the Silent mode were fatigued of audio feedback, as participant 3 said, "To be honest, I use screen readers every day and I am so sick of electronic noise."

Participants disliked Silent mode because they felt that they needed feedback to know what was happening in the application. In the words of Participant 1: "the lack of feedback makes it harder to use because you don't know whether it sees a QR code," and Participant 9: "I still prefer having more versus less feedback."

Participants that preferred the Verbal mode liked it because it provided feedback, but was less of a cognitive load than Finger Pointing. In the words of Participant 9: "the other thing I like about Verbal mode is that every time I hear a zero I think so I need to move it a little bit," as opposed to the Finger Pointing mode, which:

Presents more issues to deal with you already have to deal with how many labels are here and then I got this finger issue this finger needs to be there, but it can't be too close [and] it can't be too far away.

7. COMPARISON TO BRAILLE

While our system was designed primarily for blind users who are unable to read Braille, there are benefits for people who are Braille-literate as well.

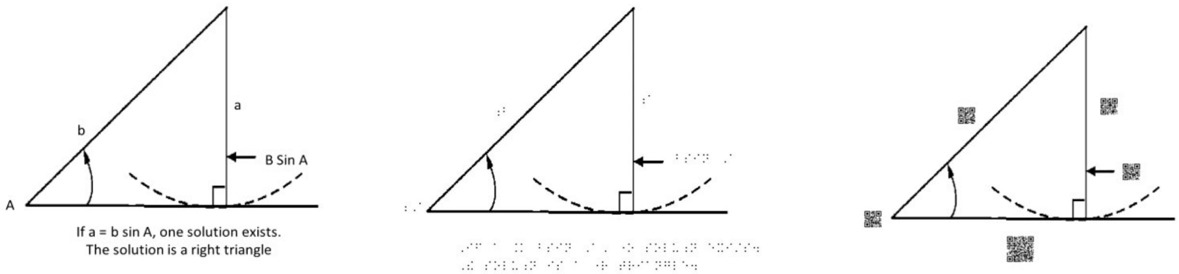

Figure 5. A comparison of the same image which is similar to one from a precalculus textbook [5] in its original form, tactile graphic form with the labels in Braille and tactile graphic form with labels as QR codes. The bottom text is a good example where the QR codes can be smaller than the equivalent Braille text.

7.1 Difficulties in Creating Tactile Graphics with Braille labels

To assess the difficulties in producing tactile graphics, we spoke with three tactile graphics experts. All three had extensive experience in creating tactile graphics and had encountered a variety of problems with the creation of tactile graphics.

From our expert interviews, one common problem was how to place Braille labels on tactile graphics. Because of the limits of human tactile perception, Braille cannot be resized to fit into a small area [3]. This means that labels with a large amount of text have to be moved. One technique for mitigating this problem is to create a key and legend. A short code is placed on the graphic where the label should be and the corresponding label is placed on a separate page. One of the experts estimated that the key and legend system is necessary for a quarter to a third of all the images he produces. Another tactile graphics expert mentioned that three quarters of tactile graphics require an explanation in order for them to be understood, and the explanation would not fit on the original graphic, requiring a second page.

7.2 Size of Braille vs QR Codes

We did a size comparison between Braille and QR code labels and found that the QR codes are able to encode 45% more text in the same amount (Figure 5). This calculation was completed by looking at 82 images from a pre-calculus textbook [5]. We calculated the estimated size of the Braille label using: the product of the number of characters in the text and the size of a Braille cell, which is the standard size of all Braille characters [3]. While many math symbols require multiple Braille characters, our conversion from text to Braille provides a good approximation. Unlike Braille, which has a standard size, QR codes vary in size based on the amount of text they encode and the distance from which they are meant to be scanned. By assuming a scan distance of six inches, we calculated the size of a QR code label based solely on the number of characters it encoded [18]. We found that the average QR code label size is 225 mm^2 and the average Braille label size is 327 mm^2.

7.3 Study

The main goal of the study was to determine if the TGV system was a feasible solution to making labels accessible to those who did not know Braille, and we feel our study demonstrates this fact. Below, we will explain the results from our comparison to Braille in the last session of our longitudinal study that was done with the 6 participants that did know Braille.

7.3.1 Accuracy

Across all participants, the average accuracy for TGV using any mode was higher than the average accuracy using Braille (Silent mode: 88%, Verbal mode: 88%, Finger Pointing mode: 89%, Braille: 77%). For the bar chart task, the TGV accuracies are similar to the average accuracy for Braille (Figure 3). While this finding goes against our hypothesis, this finding is likely due to two reasons. First, the Braille tasks the labels were written in Nemeth code. Even though we explained the how to read Nemeth-coded numbers, some participants made mistakes. Second, some of the Braille-literate participants indicated that they were out of practice reading Braille.

7.3.2 Time

The average completion time with the Braille graphics was 28.6 seconds (SD=19.0). This was faster than the average time of TGV with all of the different modes. However, after the participants learned to use the application, the times were similar. This can be seen in Figure 4, where the dot representing the Braille mode was faster than Verbal and Finger Pointing modes but slower than Silent mode in session 6.

7.3.3 Preference

Four of the six participants who used the Braille labels on the graphics stated that was their favorite. One reason was because of ease of use, as P6 stated that (with Braille): “it’s already there and you can just read it.” Additionally, people were more comfortable with Braille and thought it was more accurate. In the words of P4: *“*I’m very comfortable with Braille. It feels more reliable.”

The other two participants who preferred TGV to Braille, did not feel comfortable with their Braille literacy skills. In the words of P1:

I guess if you’re reading a textbook in Braille you’re probably up on your Braille so you wouldn’t need a smartphone or anything to access that,

and P5 said that:

I wish I had learned Braille when I was in school because that might that may have made a world of difference and I would be a lot more successful than I am right now so I do I really enjoy the Braille a lot.

8. DISCUSSION

Errors on the first three tasks (Line, Parabola, and Triangle) were a result of misidentifying which label to scan or timing out. In contrast, with the Bar Chart, errors occurred when a participant attempted to scan the correct QR code, but really scanned a different QR code. This issue occurred because of the small distances between the labels on the axes. Participants developed strategies to avoid this problem in later sessions, such as covering the neighboring QR codes. This technique helped increase the accuracy for all three modes from 55% (SD=35) for the first session to 80% (SD=30) for the last session. Figure 3 displays the changes in accuracy across the sessions by mode.

Although Finger Pointing mode was the most accurate, many participants had difficulty using the mode. If users put their finger too close to the QR code, it would not recognize the QR code. There needs to be a small gap between the finger and the QR code for both the finger and QR code to be recognized. This caused difficulties because participants did not realize that their fingers were obscuring the QR code. P2 expressed frustration:

With the pointing with the finger it kept not registering cause either my finger wasn't in the right spot or it kept picking up the wrong one somehow.

We believe this is the main reason that Finger Pointing took significantly longer than the other modes and lead to more time-outs.

9. FUTURE WORK

One opportunity for improvement is to make Finger Pointing mode easier to use. To avoid obscuring the QR code with a finger, adding tactile markers at an appropriate distance from the QR codes could make it easier for people to determine where to place their finger. Additionally, we could try using another pointer, such as a sticky note, to indicate which QR code should be selected.

The longitudinal study allowed us to assess the feasibility of TGV in a controlled setting. We look forward to see how TGV would be used in the wild. In particular, we plan to conduct a field study on the use of TGV in an educational setting.

10. ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant Nos. DGE-0718124 and DGE-1256082 and National Science Foundation Grant No. IIS-1116051. This work was supported by the U.S. Department of Education, Office of Special Education Programs (Cooperative Agreement #H327B100001). Opinions expressed herein are those of the authors and do not necessarily represent the position of the U.S. Department of Education.

11. REFERENCES

1. Aldrich, F.K. and Sheppard, L. Tactile Graphics in school education: perspectives from pupils. In *British Journal of Visual Impairment 19* (2), 69-73.
2. Bigham, J., Jayant, C., Miller, A. White, B., and Yeh, T. VizWiz::LocateIt – Enabling Blind People to Locate Objects in Their Environment. In *Proc. CVAVI 2010*, 65-72.
3. Braille Layout and Dimensions. http://dots.physics.orst.edu/gs_layout.html.
4. Elgammal, A., Muang, C., and Hu, D.. Skin detection-a short tutorial. *Encyclopedia of Biometrics*. 2009.
5. Gordon-Holliday, B., Yunker, L.E., Vannatta, G. and Crosswhite, F.J. *Advanced Mathematical Concepts, Precalculus with Applications.* Glencoe/McGraw-Hill, 1999.
6. Goransson, D. New VoiceOver Features in iOS 5. Access Lab. http://axslab.com/articles/new-voiceover-features-in-ios5.
7. Holton, B. Voiceye: A Breakthrough in Document Access. AFB AccessWorld Magazine 14, 6 (2013).
8. Jayant, C., et. al. Automated tactile graphics translation: in the field. In *Proc. ASSETS 2007,* ACM Press (2007), 75-82.
9. Jayant, C., Ji, H., White, S. and Bigham, J. Supporting blind photography. In *Proc. ASSETS 2011*, ACM Press (2011), 1-8.
10. Kane, S. K., Frey , B., and Wobbrock, J.O. Access lens: a gesture-based screen reader for real-world documents. In *Proc. CHI 2013,* ACM Press (2013), 347-350.
11. Landau, S., Holborow, R. and Jane, E. The Use of the Talking Tactile Tablet for Delivery of Standardized Tests. In *Proc. CSUN 2004.*
12. Landau, S. and Neile, J. Talking Tactile Apps for the Pulse Pen: STEM Binder. CSUN 2010 https://docs.google.com/presentation/d/1Ylscpk6QKX7Y5WCW3CFnhZDJJlL4X5ki8gHdcJvBp70/present#slide=id.i0
13. Nanayakkara, S., Shilkrot, R., Peen Yeo, K., and Maes, P. EyeRing: a finger-worn input device for seamless interactions with our surroundings. In *Proc. AH 2013.* ACM Press (2013), 13-20.
14. National Federation of the Blind Jernigan Institute. 2009. The Braille Literacy Crisis in America: Facing the Truth, Reversing the Trend, Empowering the Blind. https://www.nfb.org.
15. Nemeth Braille. http://www.braillebug.org/nemeth_braille.asp
16. Phung, S.L., Bouzerdoum, A. and Chai, D. Skin segmentation using color pixel classification: analysis and comparison *IEEE Transactions Pattern Analysis and Machine Intelligence, 27* (1), 148-154.
17. Phung, S.L.; Bouzerdoum, A.; Chai, D., Skin segmentation using color and edge information, In *Proc. of Seventh International Symposium on Signal Processing and Its Applications, 1*, (2003), 525-528.
18. QR Code Minimum Size. http://www.qrstuff.com/blog/2011/11/23/qr-code-minimum-size.
19. Rule, A.C., Stefanich, G.P., Boody, R.M. and Peiffer, B. Impact of Adaptive Materials on Teachers and their Students with Visual Impairments in Secondary Science and Mathematics Classes. In *International Journal of Science Education. 33* (6), 865-887.
20. Sheppard, L. and Aldrich, F.K. Tactile graphics in school education: perspectives from teachers. In *British Journal of Visual Impairment19* (3), 93-97.
21. Tekin, E. and Coughlan, J. A mobile phone application enabling visually impaired users to find and read product barcodes. In *Proc. Computers Helping People with Special Needs 2010.* 290-295.
22. Vázquez, M., and Steinfeld, A. Helping visually impaired users properly aim a camera. In *Proc. ASSETS 2012*. ACM Press (2012), 1-8.