Pebble Scrolling Techniques

Pavel Samsonov, Montserrat Lobos, Andrew Mittereder

May 5th, 2014

Introduction

Wearable computing in the form of smartwatches is exploding in popularity. Most of these watches have so far attempted to implement touch screen interfaces like those used on smartphones, but one – the Pebble – took a different approach. The Pebble uses a small monochrome screen as a display; it avoids the design challenge of implementing touch interactions on a watch-sized surface by using hardware buttons instead. As a result, most interfaces on the Pebble are lists of discrete menu items. Users navigate the menus by scrolling through them with Up and Down buttons, and making selections with a middle button. Unfortunately, this configuration is awkward to use, and scrolling through lists becomes increasingly challenging as they get longer. We address this issue with the introduction of two interaction techniques for scrolling on the Pebble: one relying on the Pebble's internal accelerometer to detect tilting, and the other using an external touch strip worn on the same hand as the Pebble.

Related work

Many different approaches have been taken when trying to implement innovative navigation that does not restrict screen access in small-screen devices; examples include pressure sensors along the edges of the device [10], secondary wireless input systems like a ring [3], a and a capacitive strip attached to the watch's wristband [1]. In contrast to these approaches, our proposed capacitive-strip-based interaction technique aims to explore two finger rubbing as a way of manipulating scrolling.

There has also been a lot of research on possible unconventional uses of data extracted from devices' internal sensors. Knight et al 2007 [5], Kwapisz et al 2011 [6], and Xu 2012 [7] provide a basis of knowledge on of what kind of data can be obtained from a device's built-in sensors. After exploring the range of applications for motion sensors in existing mobile devices, we began looking at how the Pebble's accelerometer could be used to enable scrolling on its small non-touch display.

Tilting as input has been around for a long time [2] and has been used as input for several different goals, but we are not aware of it being used as a scrolling technique on a wrist-based wearable. One possible reason for this may be the fact that tilting the watch by twisting your wrist eventually occludes the screen.

To understand how to match the user's scrolling intention with the device's scrolling output we will look at the transfer functions used in other devices to adjust scrolling performance. Quinn et Al. 2012 and 2013 [7, 8] offer interesting insight on the role of direction, duration, and clutching as an input.

Objective

Scrolling is the main interaction technique used on the Pebble, because its existing input provides only one-dimensional navigation. However, the configuration of this input makes navigating long lists tedious due to the slow rate of scrolling and uncomfortable positioning of the buttons. When scrolling through a music library, for instance, a user would have to take a long time to get to the song he wants to play. Thus, we wanted to create a better scrolling technique that allowed users to scroll more quickly, with less physical effort or discomfort than currently required by the Pebble's buttons

We also wanted to make this technique one-handed. Currently, the Pebble requires both arms to use - while the hand on one arm manipulates the buttons, the other does nothing but wear the watch. Since wearing the watch is an unavoidable requirement, we wanted to transfer the manipulation functions to that arm and free up the other one for interactions with the environment or other devices.

Design Process

Tap Scrolling
We started our design process by brainstorming different ideas for scrolling techniques, using the Pebble's existing capabilities as a starting point. We selected the Pebble's internal accelerometer as the most promising sensor for our scrolling technique. Our first idea was to scroll and select items by "tapping" the Pebble in different directions. This functionality is built in to the Pebble's SDK, so we thought it would be easy to implement. We prototyped a Pebble application that responds to tap events sent by the Pebble API. Taps in the Y-axis (up and down parallel to the watchface) scroll up or down, and taps in the X-axis (left and right parallel to the watchface) select the current item or go back. This technique was not successful, because taps were very hard to trigger, requiring a large amount of force. When a tap was triggered, the intended direction of the tap often did not correspond to the direction actually recognized.

To address this, we created a second version of the app that manually recognized taps instead of using the built-in tap recognizer. This was done by reading accelerometer packets in groups, taking the average reading, and then comparing it with a calibrated baseline. If the difference in a certain axis surpassed a threshold, a directional tap was recognized. This version of the app was more sensitive and consistent than the previous approach, but it was very slow to respond to taps;

it would typically take about a second for the selection to change after a single tap. We concluded that the tapping approach was not practical within the timeframe of our project. We briefly considered prototyping a tapping approach using a smartphone with a better accelerometer than the Pebble's, but thought we should try other approaches on the Pebble first.

Tilt Scrolling

Our second accelerometer-based approach was to tilt the watch in different directions to scroll. We believed this technique would have potential, because it could allow for multiple scrolling speeds depending on how far the Pebble is tilted. We created a rough prototype Pebble app that responds to tilts of the wrist to scroll up and down. We found scrolling to be much more consistent and sensitive compared to the tapping approach. Our first prototype had issues with response times, but this was mitigated by increasing the sampling rate of the accelerometer.

In an informal heuristic evaluation performed on the researchers, we found that tilting the wrist is straining, and also found that it's hard to see the Pebble's display when turning the device away from you. To address this, we tried incorporating a physical modification to the Pebble by adding a convex hemisphere-shaped curve to the back of the watch. This allowed tilting by pressing down on the sides of the watch but restricted the interaction by requiring the user to use both hands. Although scrolling by pressing the edges had the potential to work efficiently, the cost of implementation did not justify the interaction technique. Changing the layout of the buttons or simply adding a touchscreen would achieve the same goal. With this in mind, we decided to focus our research in one-handed interactions with the Pebble.

Ring Scroll Wheel

The notched mouse wheel is currently a very common scrolling technique on conventional computers. We wanted to adapt it to the Pebble for two reasons: the Pebble's menus are all linear lists that are scrolled in increments of one, and the notched mouse wheel had already proven to be a very fast and accurate scrolling technique in user tests. We envisioned 3D printing a ring with a wheel that would be worn on a user's index finger and operated with the thumb of the same hand. Interlocking gears would transfer the motion of the ring to a shaft encoder. However, this concept proved impractical for two reasons. First, the encoder setup we planned to use turned out to be too complex for 3D printing. Second, the encoders we tested proved to be very inaccurate - notched encoders had too few steps per revolution to make scrolling on a small thumbwheel effective, and smooth encoders made accurate selection difficult when tested in a preliminary trial. As a result, we decided to proceed with an input device that required no moving parts at all and could be as accurate as we wanted: capacitive sensing.

Additionally, we were not able to successfully pair the Bluetooth Arduino to the Pebble – the current software on the Pebble does not officially support such a connection, and the Pebble development community has not yet been able to find a work-around. Since these hardware input devices did not require anything from the Pebble other than its screen, we decided to pair the

Arduino to a computer and port our testing program to it. Faster or more accurate results than the incumbent would prove that the capacitive scrolling technique is worth implementing on the Pebble hardware as part of future work.

Scrolling Techniques

Based on our design analysis presented above, we decided to move forward with testing the following three techniques.

Technique 1: Pebble buttons (baseline)

Figure 1. Using the Pebble's buttons to scroll.

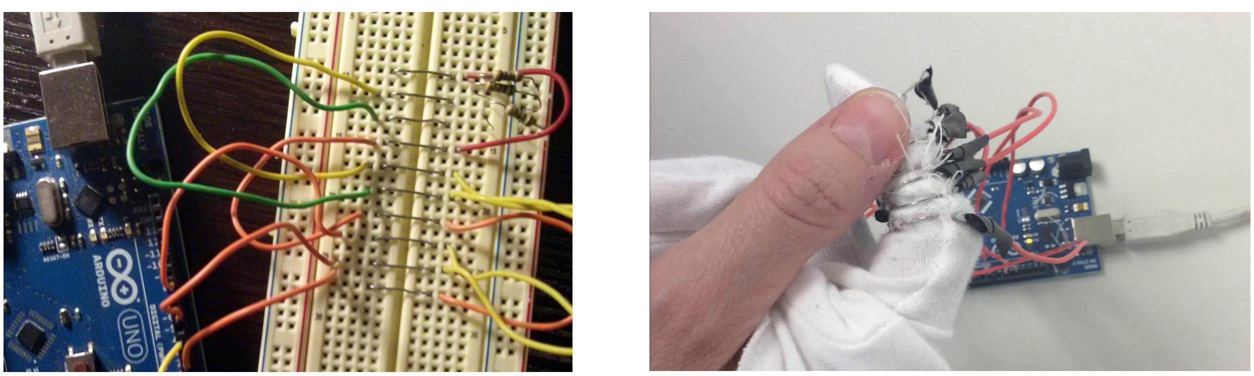
The current method of scrolling and selecting items on the Pebble is with three buttons on the right side of the device. The top and bottom buttons scroll up and down, respectively. Pressing and releasing a button scrolls one increment, while pressing and holding a button scrolls continuously. After holding the button down for about one second, the rate of scrolling increases. Selecting an item is done by pressing the middle button. The Pebble used for testing was running version 2.0 of the Pebble firmware.

Technique 2: One-Handed Pebble tilting

Figure 2. Scrolling through a list using the Tilt Interaction.

We decided to test one-handed tilting, hoping to incorporate tilt-scrolling as a way to augment the current scrolling capacity of the Pebble. We suspected that tilting was an effective way of scrolling, especially throughout long lists.

Our implementation allowed the user to scroll up or down one item in the menu by briefly twisting her wrist in that direction. Holding the wrist in the twisted position allowed the user to scroll in that direction by 2 items per second. Twisting the wrist further increased the scrolling speed to 10 items per second (equivalent in speed to the fastest scrolling mode of the incumbent button technique). After 2 seconds of being held in this position, the speed of the scroll increased to 20 items per second. After navigating to the desired item, the user selected it by moving her forearm up or down. We added a two second timeout after the choice is made to prevent accidental selections.



Technique 3: Capacitive strip on a glove

(a) *(b)*

Figure 3. Capacitive strip prototype. (a) Breadboard prototype. (b) Strip mounted on a glove.

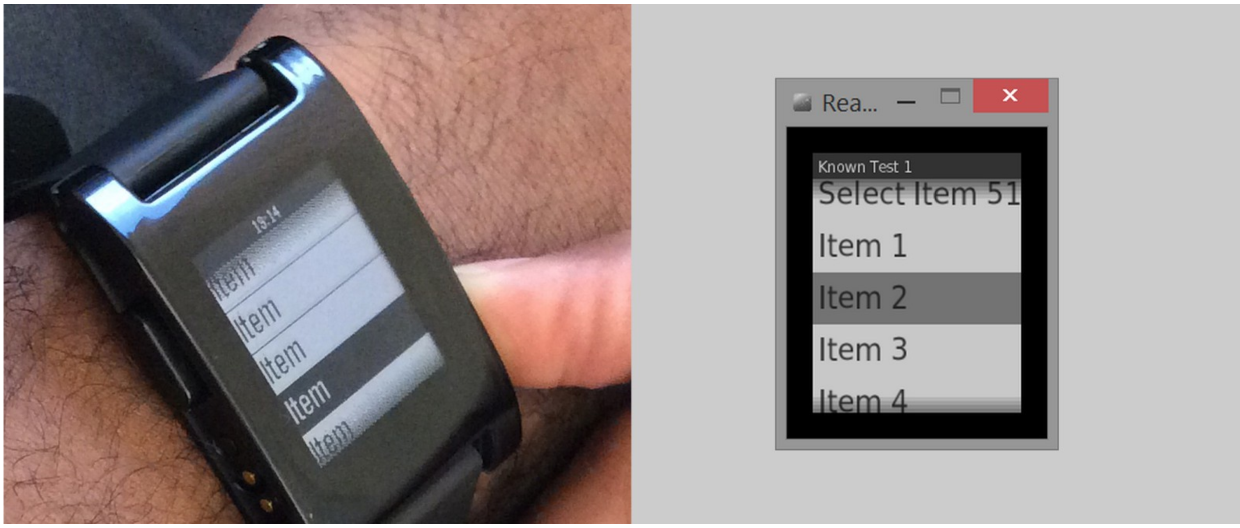
The capacitive strip device we developed consisted of a series of 5 parallel wires, 10 mm in length and spaced 7 mm apart. It used the CapSense library to detect when the user touched one or more of these wires. We left the wires uncovered, which allowed them to act as notches that provided 1:1 physical feedback to the user when scrolling through itemized lists.

The wires were sewn to a cotton glove, along the side of the index finger. The wires were bent in a semicircle to fit the shape of the finger, but were not connected to one another by any rigid materials. This allowed the user to bend her finger into a comfortable position for scrolling. The user scrolled by swiping her thumb along the index finger, and made selections by tapping the scroll surface without swiping. This technique does not require the use of the arm not wearing the Pebble, but because most people wear watches on their non-dominant hand, we expect some reduction in accuracy.

User Testing

Participants
We performed a between-subjects evaluation of the three scrolling techniques described above. This allowed us to perform user-testing in parallel to developing the prototypes. We tested 6 users with the baseline technique, 5 users with the tilt technique, and 3 users with the glove technique. Participants ranged in age from 18-44 years old, and were all students at Carnegie Mellon University. While all participants were proficient with technology, few of them have ever used smartwatches.

Test Applications
To test both the baseline and tilt techniques, we developed a Pebble application that records the users' performance by measuring their speed and accuracy. We also implemented a version of the app in Java using Processing to test the touch strip interaction, because it needed to be connected to a computer. This app imitated the Pebble's user interface so that results could be meaningfully compared to the baseline.



(a) (b)

Figure 4. Test applications. (a) Pebble application to test baseline and tilt. (b) Processing application to test glove.

The test app included two sections: one for known-target finding and a second one for unknown-target finding. For known-target scrolling, participants were asked to look for a number in a numbered list; the number changed for each trial. For unknown-target scrolling, participants were asked to search through a list and find the item labeled "*****NEXT" without knowing which position it was in. Once the participant made a selection, the app recorded both the time it took for the participant to select the target and the accuracy of the selection. Accuracy was determined by assigning a value of +1 to every list item past the target and -1 to every list item above the target. One of the researchers observed each test in order to note any backtracking and corrected overshooting behaviors, which the app did not capture.

(a) (b) (c)

Figure 5. Screenshots of scrolling test app. (a) Known trial. (b) Unknown trial. (c) Results page.

We explained the task to participants before starting, and there was a practice run before each section of the test starts recording any data. Each round of the test included 10 trials for each target type. We ran 3 rounds of tests per participant in order to observe any potential learning or fatigue effects.

Environment

The test was conducted in a facility that allowed users to be seated and rest their arms on a desk. Users were asked to find a comfortable position before starting and to keep it consistent throughout the test. Participants wore the Pebble on their non-dominant hand. After going through the 3 rounds of testing, the participants filled out a survey to assess their familiarity with technology and their impressions of the task.

Results

We first present the overall time and accuracy of each scrolling technique across all users (Table 1). This is presented as the average total time to complete the scrolling test in its entirety, along with the percentage of trials where the assigned target was correctly selected. Comparing the tilt scrolling technique to the baseline, we see that it took significantly longer for users to complete the test, and was also significantly less accurate, with only 67% of trials selected correctly. The glove, on the other hand, was slightly faster than baseline (though the difference does not appear to be statistically significant). It was also 15% less accurate than baseline, with 83% of selections being correct.

Technique	Average Total Time	Percentage Correct
Baseline	211.61 seconds	98%
Tilt	249.31 seconds	67%
Glove	195.88 seconds	83%

Table 1. Overall time and accuracy of the scrolling techniques.

We performed further analysis on “known” vs. “unknown” scrolls. Known scrolls correspond to trials where the user had to find a specific item, while unknown scrolls correspond to trials where the user had to find “*****NEXT”. Comparing tilt to baseline, we see that tilt scrolling was significantly slower across both known and unknown, taking ~20 seconds longer for each. Glove scrolling performed at a similar speed to baseline on known scrolls, but was ~15 seconds faster on unknown scrolls. Another interesting finding is that while baseline took ~10 seconds longer for unknown scrolls compared to known scrolls, the glove’s total time for each was about the same. This shows that the glove is a versatile scrolling technique that works well regardless of the type of scroll.

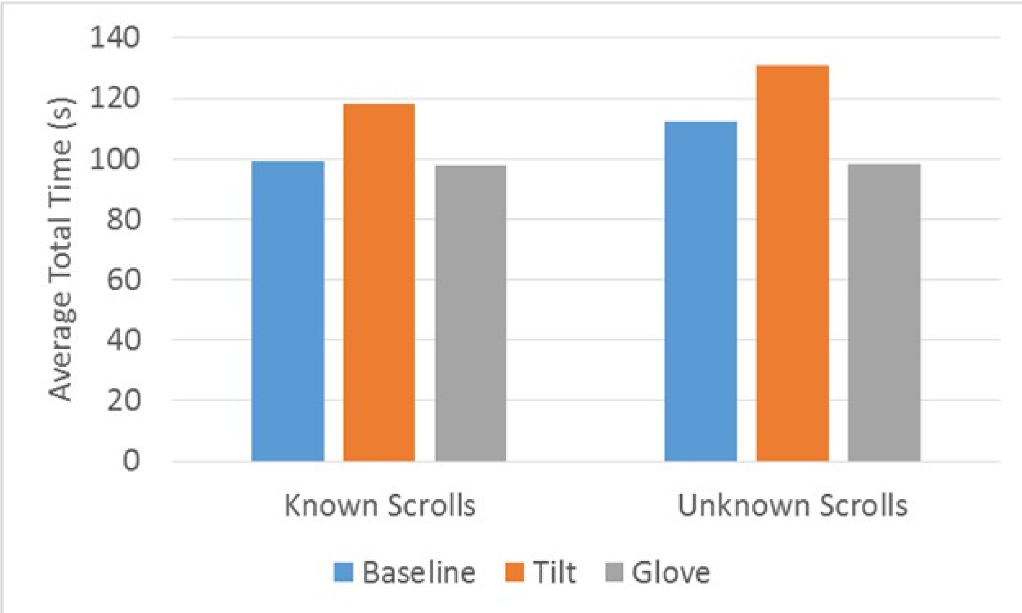


Figure 6. Average time for known vs. unknown scrolls.

We also compared "short" vs. "long" distance scrolls. Short scrolls were defined as trials where the user had to find an item at position 25 or smaller, while long scrolls were defined as trials where the user had to find item at position 26 or larger. Again, we see that tilt scrolling was significantly slower than baseline across both types. The glove, while slightly slower on short scrolls, was slightly faster on long scrolls. This suggests that the glove can speed through a list more quickly than baseline (whose speed is capped when holding the button down).

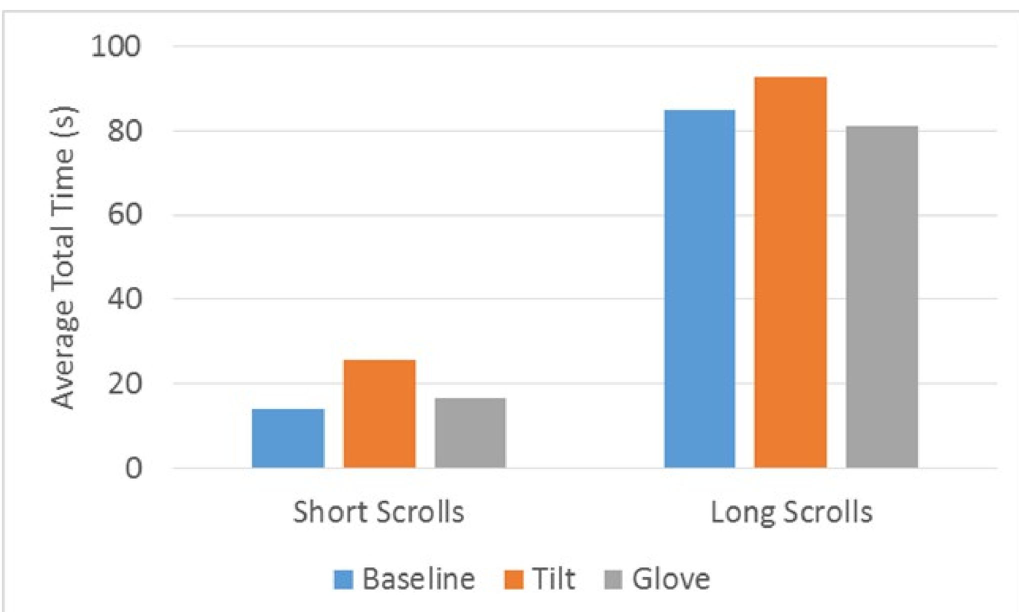


Figure 7. Average time for short vs. long scrolls.

Lastly, we evaluated user preferences for the scrolling techniques through a short questionnaire given to participants at the end of the study. With the baseline technique, all participants gave it an average rating, describing the experience as either “It was ok” or “I had some problems.” Users found the buttons to be tiring and difficult to press, making comments like “The button was difficult to press, needed to counter tension with my thumb, forcing me to squeeze the device.”

All five users that tested the tilt technique expressed difficulty with using it. Two users rated the experience as “very hard,” while three users wrote “I had some problems.” Some comments from participants include: “Fine control was difficult. I had to go back and forth several time to make each selection” and “It was difficult to tilt my wrist. Specially inwards.” All users also noted pain or discomfort in the wrist and shoulder at the end of the test.

Glove users responded quite differently from tilt. Two users rated the experience as “very easy”, with the last user rating it as “It was ok.” Comments about the experience include: “I think it was fun and easy” and “When it worked it was really cool, but when it didn't, it was annoying.” The participants noted some fatigue at the end of the test, but no pain or discomfort.

Discussion

About the incumbent

When using the Pebble buttons to do long scrolls, all participants overshot by ~2 items when not using discrete pushes. However, buttons provided a very accurate way of correcting overshooting.

Although most participants were successful in targeting long scrolls, many of them complained about the task taking too long and the buttons being hard to push. For example, one participant commented that “it was long! If it is a smartwatch it should be smart enough to skip to the end.” Users also found the task surprisingly straining.

Ergonomic issues of the tilting technique

The biggest challenges we observed in the tilt interaction were backtracking and selection. Although correction was easy for participants using the Pebble’s buttons, when using tilting backtracking turned out to be very hard. Discrete wrist-nudging was not easy and not always effective, especially when participants tried to scroll up by increments of one. Most times when trying to go back by small increments users ended up overcorrecting.

All of our participants reported some kind of physical discomfort or strain. Many of them manifested troubles keeping the target in sight when tilting their wrist away from their body. In general, they mentioned going down being easy but going up being much harder.

Selecting was also a pain-point. Once participants reached their target, often times the vertical movement intended for selection ended up scrolling away from the target before applying the selection or not selecting at all. Moreover, selection was sometimes activated ‘randomly’ in the middle of a scroll. We observed users performing some small movements to release tension in their arms, which sometimes triggered the selection command even though the user’s intention was not to select but to keep on scrolling.

One participant mentioned lack of feedback being a problem. He said it was hard to determine whether a selection had been executed successfully or not. Moreover, the vertical movement required for selection created some noise in the opposite axis, resulting in some unwanted scrolling; this caused the instructions for the next trial to move out of the screen, aggravating the users’ confusion on whether the item was successfully selected or not.

General impressions on the glove

Although all three participants found the glove to be easy to use, two of them starting scrolling in the wrong direction on their first trial. This suggest that we should further test a larger user base to determine which direction feels more natural to the general public.

Users mentioned some numbing on their fingertips after the tests, which is probably caused by the prototype’s low fidelity and the solder’s quality.

Undershooting and overcorrecting using the glove:

Scrolling in discrete increments worked generally well due to the notched nature of the glove’s capacitive strip, except when users are starting a new movement. In this case, scrolling gets mixed with tapping. Users tend to do a long scroll to ‘activate’ the technique. They seemed hesitant about how to approach this task.

There was definitely a problem when users reached the end of the strips and were short of their goal by 1 item. When participants tried to correct by one increment, they tended to scroll by at least 2 contacts.

Accuracy in long and unknown target position scenarios

We speculate that the improved performance of the capacitive scrolling glove in long and unknown scroll tasks is due to the difference in how users activate the technique. Both the incumbent and tilt techniques use a rate-controlled scroll that requires the user to hold a position when they want to continue scrolling. This strategy works well when the user knows the target number, because they can prepare to let go ahead of time. When the target is unknown, however, the user does not have forewarning of when to let go, and ends up overshooting and having to backtrack. The glove instead relies on the user repeating a scroll action that advances slightly more than one screen every complete swipe. No scrolling occurs while the user’s thumb moves from the end of the capacitive strip to the beginning, giving the user a moment to assess the content on the screen and spot the target if it has appeared.

In addition, the incumbent scrolling method demonstrated faster performance in the short scroll trials. This is likely because the fast scroll speed does not activate until the user has been holding down the button for a little while, resulting in a slower speed when the user encounters the target, and therefore a slower required reaction time to stop the scroll. For the long scroll tests, the user was always in fast scroll mode, and needed a faster reaction time to stop without overshooting. The glove, on the other hand, has the same scrolling speed regardless of how long it was used for, which gives it an edge in long scroll tasks when compared to rate-controlled techniques, but provides no benefit in short scroll tasks where the rate-controlled techniques are still moving slowly enough for the user to react in time.

Learning effects and participant strategies

Although our results regarding learning effects and fatigue are inconclusive, we did notice participants developing strategies to improve their performance or reducing fatigue while running the tests. In the second round of testing using the tilting technique, participant 5 discovered that knocking down on the table was a more effective way of selecting than trying to move his arm up. Participant 5 also tried holding his arm up above his head and holding it with his non-dominant hand to overcome fatigue. About half-way though the first run of testing, Participant 2 placed her fingertips on the table with her wrist lifted and dropped her wrist down whenever she needed to select an item. She verbally expressed that this strategy seemed to work for her pretty well; in fact, she prided herself on ‘mastering’ selection.

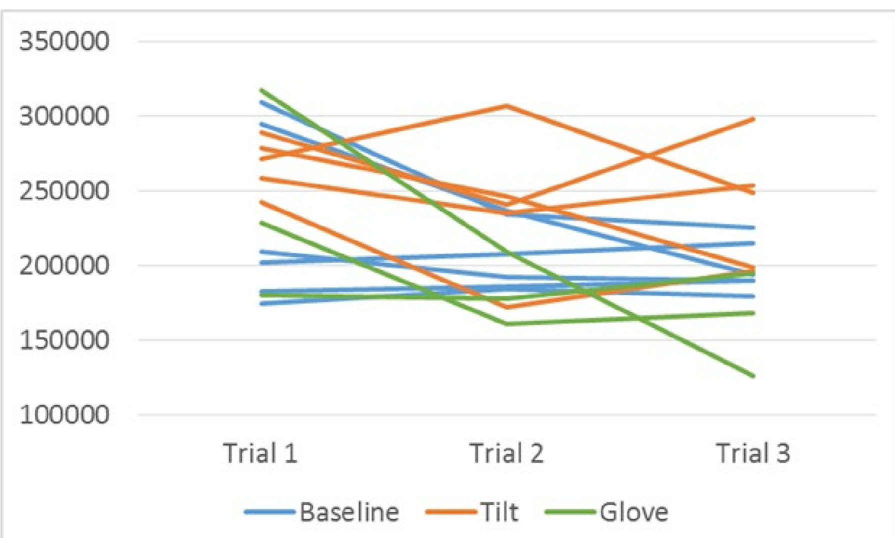


Figure 8. Time performance across trials and techniques

Looking at the chart we can notice a pattern in every user’s performance but one: they improve their time-performance from trial 1 to trial 2, but time goes up again in the third trial. Based on the comments we received from our participants, we believe that this may be caused by fatigue. Further testing with longer trials may be helpful to come to a conclusion on this issue.

Conclusions/Future Work

Overall, we can conclude that the tilt scrolling technique is a step backwards compared to the baseline. It was found to be significantly slower, less accurate, and rated poorly by users. It was also found to be painful and uncomfortable, with fatigue effects overpowering learning effects in the study. It is clear that the technique requires more tuning and design iterations. For example, we should try to calibrate forward movement to be more sensitive, because most of the users reported inward torque as being more difficult. Accidental selection was also a huge problem with this technique, and should be addressed.

The capacitive glove is a promising scrolling technique for the Pebble. It was found to be as fast or faster than the baseline (depending on the type of scroll trial), while only being slightly less accurate. Future work for this technique includes more tuning of the scrolling algorithm, along with experimentation with the number and spacing of contacts: the breadboard prototype used 10 contacts and demonstrated faster scrolling speeds, but was too long to mount on a finger. Page up/down gestures or inertial scrolling could also be implemented to alleviate fatigue.

There is also room to improve the user testing framework for future studies. In particular, the test should be able to capture the amount of overshoot and compensation of participants. One simple way to do this would be to capture how many times the user passes by the target in either direction before selecting it.

References

[1] Daniel Spelmezan, Caroline Appert, Olivier Chapuis, and Emmanuel Pietriga. 2013. Side pressure for bidirectional navigation on small devices. *In Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services* (MobileHCI '13). ACM, New York, NY, USA, 11-20.

[2] Jun Rekimoto. 1996. Tilting operations for small screen interfaces. In Proceedings of the 9th annual ACM symposium on User interface software and technology (UIST '96). ACM, New York, NY, USA, 167-168. DOI=10.1145/237091.237115

[3] Daniel Ashbrook, Patrick Baudisch, and Sean White. 2011. Nenya: subtle and eyes-free mobile input with a magnetically-tracked finger ring. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '11). ACM, New York, NY, USA, 2043-2046.

[4] W. Buxton and B. Myers. 1986. A study in two-handed input. *SIGCHI Bull*. 17, 4 (April 1986), 321-326

[5] James F. Knight, Huw W. Bristow, Stamatina Anastopoulou, Chris Baber, Anthony Schwirtz, and Theodoros N. Arvanitis. 2007. Uses of accelerometer data collected from a wearable system. *Personal Ubiquitous Comput.* 11, 2 (January 2007), 117-132

[6] Jennifer R. Kwapisz, Gary M. Weiss, and Samuel A. Moore. 2011. Activity recognition using cell phone accelerometers. *SIGKDD Explor. Newsl.* 12, 2 (March 2011), 74-82.

[7] Zhi Xu, Kun Bai, and Sencun Zhu. 2012. TapLogger: inferring user inputs on smartphone touchscreens using on-board motion sensors. *In Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks* (WISEC '12). ACM, New York, NY, USA, 113-124.

[8] Philip Quinn, Andy Cockburn, Géry Casiez, Nicolas Roussel, and Carl Gutwin. 2012. Exposing and understanding scrolling transfer functions. *In Proceedings of the 25th annual ACM symposium on User interface software and technology* (UIST '12). ACM, New York, NY, USA, 341-350

[9] Philip Quinn, Sylvain Malacria, and Andy Cockburn. 2013. Touch scrolling transfer functions. In Proceedings of the 26th annual ACM symposium on User interface software and technology (UIST '13). ACM, New York, NY, USA, 61-70.

[10] Simon T. Perrault, Eric Lecolinet, James Eagan, and Yves Guiard. 2013. Watchit: simple gestures and eyes-free interaction for wristwatches and bracelets. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 1451-1460

Appendix

Tilt/Glove Test Protocol

The test should be conducted in a facility that allows users to be seated and rest their arms on a surface. Users will be asked to find a comfortable position before starting and to keep it consistent throughout the test.

Participants should be wearing the Pebble (or the glove) on their non-dominant-hand wrist.

After going through the 3rd round of testing, the participant should fill this survey.

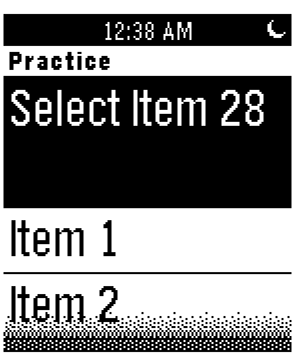
Explain the task to participants:

You will be asked to find a target using this interaction technique.
Each round of the test takes ~4 minutes and we need you to do this 3 times.
You can take a break of 1 minute between each round.
The whole test should take ~15 minutes total.

Before starting, please find a comfortable position and try to maintain it throughout the test. At least your elbow should be resting on a flat surface so you don't need to move your upper arm.

You will be asked to perform two different tasks:

- Finding a number in a list
- Finding the word "*****NEXT" in a list



The instruction specifying the target you need to look for will be displayed at the top of the screen in each trial.

The first two trials are for you to practice and become familiar with the technique.

How to operate the Tilting technique:

- Nudge your wrist up or down to scroll one item (if you hold the nudge you can scroll slowly, 2 items/second)
- Tilt your wrist farther to start scrolling fast (10 items/second, as fast as pressing and holding the buttons)
- If you hold a fast scroll for >2 seconds, it will enter a “hyper-scroll” mode, scrolling 20 items/second
- To select an item, move your forearm up or down (keeping your upper arm still). Make sure to move your arm back to the resting position, because it will select again if your arm is still tilted 2 seconds afterward.

How to operate the Glove technique:

- Wearing the glove on your non-dominant hand, place your hand as if you were looking at your watch-face. Try to maintain that position throughout the test.
- To scroll, slide your thumb across the capacitive band placed along the side of your index finger.
- Tap on the band to select an item.