Comparison of Push Techniques for Cross-Device Interaction Between Mobile Devices and Large Displays

Bjarke M. Lauridsen
Aalborg University
blauri10@student.aau.dk

Ivan S. Penchev
Aalborg University
ipench14@student.aau.dk

Elias Ringhauge
Aalborg University
eringh10@student.aau.dk

Eric V. Ruder
Aalborg University
eruder10@student.aau.dk

ABSTRACT
In recent years, research into cross-device interaction techniques has increased. Much of this research focuses on interaction between mobile devices and large displays. We contribute to this body of knowledge with an empirical comparison of four different push techniques - *Pinch*, *Swipe*, *Throw*, and *Tilt* for interaction between mobile devices and large displays. We report on success rate, efficiency and accuracy. We also present the ease of use of techniques as perceived by users. We show that *Swipe* was the most effective in terms of success rate, efficiency and accuracy. Furthermore, *Swipe* gathered the highest score, in regards to ease of use, by users. Participants also reported that *Pinch* was the most fun to use.

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous; See http://acm.org/about/class/1998/ for the full list of ACM classifiers. This section is required.

Author Keywords
Interaction Techniques; Cross-Device Interaction; Natural User Interaction; Kinect; Mid-air Gestures; Smartphones; Large Displays; Data Transfer

INTRODUCTION
The evolution of ways people interact with the digital world is noteworthy considering the short life-span of computing. How we use our devices, which devices we use, and the context in which we use them has been continually under transformation. From portable personal computers originally considered mostly for specialized field applications such as accountancy, military use, or for sales representatives, which addressed mobility of a person's workspace, to modern hand-held devices which presents their users with such degrees of freedom that ultimately workspaces are becoming more ubiquitous. In non-work context people are now connected mostly everywhere, which aids us in a search for information or in communicating, changing the way we interact. This expansion has not only increased mobile computing due to greater convenience, but also made it widespread.[6]

As numerous divergent devices are being adopted in different domains and contexts, understanding cross-device interaction is currently becoming more important and relevant; after all, people take their hand-held devices into situations where other technologies are active. This ubiquitous presence of devices means that they can be used to enhance everyday situations in all kind of places. Imagine if a public display could morph from a one-way broadcast device that merely shows visual content to a two-way interaction device that provides a more engaging and immersive experience. This emergence of cross-device communication opportunities prompts a need to understand how different interaction techniques perform in use, i.e. in terms of how easily, quickly and accurately, or in terms of how enjoyable or satisfying it is to interact in this way.

Research in the area of cross-device interaction is increasing with the changing trends. Earliest examples are in the late 90's, within ubiquitous computing, with Rekimoto's work. He argued for what he called multi computer user interface and that interaction techniques must overcome the boundaries among devices in multi-device settings[16].

Recent HCI research has focused on how to include natural modality more in cross-device interaction, contributing to what should be know as cross-device natural user interaction. Some researchers used spatial information [11, 12], others used touch [19], or combined touch with air gestures [4]. But we still have limited understanding of how to design cross-device natural user interaction techniques and we lack empirical studies of this.

Inspired by the opportunities presented by such challenges, this paper reports on a empirical study between four different cross-device natural user interaction techniques for data transfer between mobile devices and large displays. We discovered that out of the four techniques we developed and implemented, *Swipe* was the most effective technique. We also show that even though *Pinch* is not as effective as the others, users described it as a fun technique.

RELATED WORK

Public displays are an inherently visual medium with graphics and animations that are increasingly used to visualize data. However the general case is that presented data can only be viewed and not interacted with. The current trend is a change from non-interactive information broadcasters to active medium information exchange. Combining this with the recent trend of increased number of hand-held device per capita an opportunity for new cross-device applications in public space arises, providing new opportunities for exploring cross-device interaction (CDI), in this context.

In a historic perspective, one of the earliest working cross-device applications is by Myers et al. [13]. One of the applications realized within their *Pebbles project* is SlideShow Commander that utilized Personal Devices Assistants (PDA) to control a PowerPoint presentation running on other computer or laptop. It was possible not just moving between slides, but also scribbling and writing on the PDA slides, while annotations are shown on the presentation for the audience. However the idea of cross-device has deeper roots in ubiquitous computing. Rekimoto's work on *pick-and-drop technique* is one of the earliest examples for exploring a technique that spawns between multiple devices. The technique " *allows a user to pick up an object on a display and drop it on another display as if he/she were manipulating a physical object.*" [15]. These two early work examples, even though in different fields, are related and would provide a foundation for additional research.

An example of such research is that of Boring et al. [2] who not only build a cross-device application but explored the implications of different techniques on it. Boring et al. explored the transfer of data from a large public display onto a mobile device. They created a method of transferring data from a large screen by using the camera on the mobile device. The user would take a picture of whatever content they were interested in, after which the application would query the content server with the picture taken from the user. Through visual analyses, the content server would determine what content the user was interested in and would return that content to them. They show that there is a need for enabling data exchange between mobile devices and public displays. In another study, Boring et al. [3] investigated cross-device interaction between large displays and mobile phones. Investigating three different interaction techniques in order to continuously control a pointer on a large screen from a mobile device. *Move* and *Tilt*, two of the three interaction techniques, enabled faster selection time compared to the last one, *Scroll*, but at the cost of higher error rates. They showed that different interaction techniques have certain strengths and weaknesses, and depending on the context and use, certain techniques are more effective.

Boring et al.'s idea [3] of how to control a public display is only one side of cross-device interaction, a different idea from Nielsen et al. [14] uses collaboration surface made from multiple devices to investigate the use of multiple device together, by allowing a number of devices being put next to each other and " pinched " together to form a larger collaborative workspace. In order to expand on the idea of common workspace, a movement from use of multiple devices to build one large display was needed. Schmidt et al. [18] proposed a cross-device interaction style for mobiles and surfaces where one can use multiple phones to interact with a digital surface.The researchers point out that *"natural forms of interaction have evolved for personal devices that we carry with us (mobiles) as well as for shared interactive displays around us (surfaces) but interaction across the two remains cumbersome in practice"*. In order to overcome this they propose the use of mobiles as tangible input on the surface in a stylus like fashion.

A combination of the ideas is presented by Skov et al. [20] illustrating six different cross-device interaction techniques for the case of card playing. A player can see their own cards on their phone and use three different techniques for playing a card from the hand-held to the tablet, which is placed on a table. In the other direction, i.e. when drawing a card, the player also has three techniques to choose from. The study aims to quantitatively evaluate each of the techniques and shows that there is a difference in time and number of errors between the techniques. They recorded two types of errors, namely, interaction errors and play errors. The number of interaction errors shows how difficult it is for a user to perform a given technique while play errors represent the errors related to the game and is recorded when the user plays a wrong card. The difference in interaction errors is apparent, especially between two of the techniques for playing a card.

The research above illustrates two different direction movement in cross-device interaction, what they have in common, is data transfer between devices. Hamilton and Wigdor's work [7] aggregates much of the works above and clearly articulates the data transfer. They create a prototype framework for cross-device applications by combining a number of interaction techniques for data transfer, chaining tasks, and managing interactions sessions. Data transfer is a challenge and as such there are different approaches for solution. Marquardt et al. [12] study cross-device interaction on tablets with a natural modality, by involving spatial information through proxemics.Based on the constructs of f-formation, micro-mobility, and co-present collaboration, they build their prototype with the idea of support for fluid and minimally disruptive interaction in document transfer. Bragdon et al.[4] propose using a combination of air gestures and touch.They aim to design, implement and test a system that allows, a group of users, to interact using air gestures and touch gestures. The purpose is to increase control, support democratic access, and share items across multiple personal devices such as smartphones and laptops where the primary design goal is fluid, democratic sharing of content on a common display.

INTERACTION TECHNIQUES

In this section we illustrate and describe techniques that can be used to push information from mobile devices to large displays with the purpose of empirically comparing them to each other. We want to find intuitive techniques that allow a user to walk up and use a large display. The techniques are characterized by allowing the users to interact with a large screen in a natural way using one or both hands and their mobile phone.

All the techniques were found in the literature and chosen based on a set of criteria outlined in Table 1. We are also interested in examining the effect of each technique on targets of two different sizes, large and small, relative to the size of the screen because the need for precision of an application is not always the same. Sometimes an applications needs to allow the user to place some data in really specific locations, where sometimes a general approximation is enough. We chose four techniques named *Pinch*, *Swipe*, *Throw* and *Tilt* because they fulfill different aspects of the criteria requirements in Table 1 and allow us to compare them to each other in an experimental setup with simple tasks.

Criteria	Description
Number of hands	There must be both one-handed and two-handed techniques.
Previously used	To avoid designing and testing a set of novel techniques, we had the criterion that all techniques must have been used by others before we would use them.
Complexity	The techniques must differ in their complexity and therefore we included techniques with different amount of steps.
Natural feel	There must be a natural feel to the techniques in some way.
Time	The time it takes to perform the different techniques must be different.

Table 1. This table describes the set of criteria.

In the following we explain why these four techniques were chosen and how they should be performed.

The *Pinch* technique (Figure 1) is used in [8] by Ikematsu et al., as part of a drag-and-drop method for moving data objects between devices. Chen et al. uses a pinching gesture in [5] for cross-device interaction between a smartphone and a smartwatch to control volume. Benko and Wilson [1] used the *Pinch* technique for interacting with omnidirectional visualizations in a dome. This technique is again a combination of the pointing technique used by Scheible et al. [17] and the aforementioned pinching techniques. The reason for including this technique was to imitate the natural action of picking up a real object e.g. piece of paper, and then moving it to another location. With *Pinch* we get a two handed technique which requires the user to perform a series of steps and are thus considered more complex and time consuming than the one handed *Swipe* and *Tilt*. The *Pinch* technique is performed by *1*) holding the phone in one hand and making a pinch gesture on the phone with the other hand (fig. 1a), subsequently closing the hand; *2*) pointing at a target on the large display with the closed hand (fig. 1b); and *3*) opening the hand to complete the technique (fig. 1c).

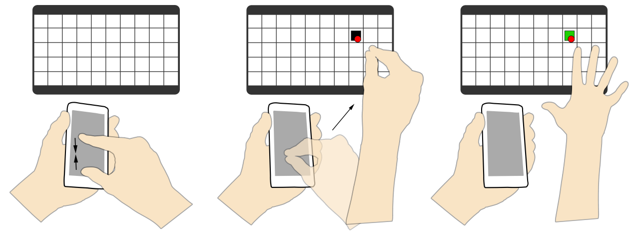


Figure 1. (a) First step of the *Pinch* technique is to make a pinch gesture on the phone using the index finger and thumb. (b) Second step is to move and use the pinched hand as a pointer. (c) Third step is to release by opening the hand.

The *Swipe* technique (Figure 2) is used by Bragdon et al. [4] in Code Space, a system using the Kinect and smartphones to support developer meetings. Bragdon et al. describe the technique as: *"cross-device interaction with touch and air pointing"* and the swipe motion is described as *"flicking up on the touch screen"*. This technique was chosen because of its simplistic design and the low level of complexity. Only one hand is needed and the amount of effort and time required to execute this technique is minimal compared to other techniques. *Swipe* is copied exactly as described in [4]. The *Swipe* technique is performed by *1*) pointing at a target on the large display with the phone in a stretched arm, and *2*) making a forwards swipe motion with the thumb on the phone's screen (fig. 2).

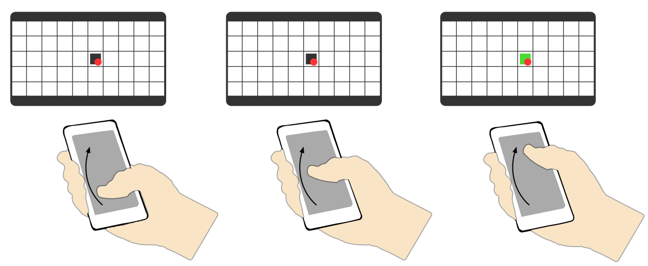
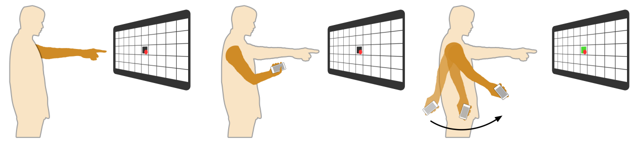


Figure 2. The *Swipe* technique is performed by using the thumb to swipe from (a) to (c).

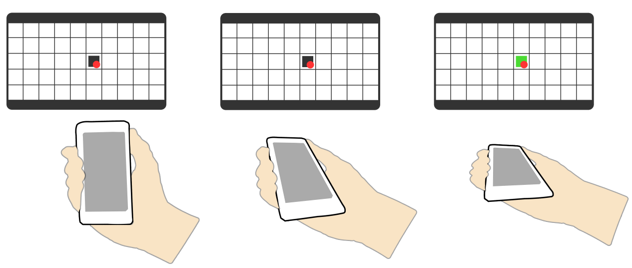
The *Throw* technique (Figure 3) is a combination of a technique for pointing [17] i.e. using a hand as a cursor in mid-air, and a throw technique described by Walter et al. [21] used in a system for sharing information on large public displays. We chose to include this technique based on its natural and playful design. The technique mimics the real world scenario of throwing something like a ball somewhere or to someone. *Throw* is two handed, more complex than aforementioned techniques, and takes a little longer to execute because of the increased number of steps. The *Throw* technique is performed by *1*) pointing at a target on the large display with one hand (fig. 3a); *2*) holding the phone in the other hand and selecting data (fig. 3b); and *3*) making a swinging motion towards the large display (fig. 3c).



(a) (b) (c)

Figure 3. (a) First step of the *Throw* technique is to point at the screen. (b) Second step is to select data. (c) Third and final step is to do a swinging motion with the phone towards the screen.

The *Tilt* technique (Figure 4) is used in a collaborative application by Lucero et al. [9] to transfer an object from a large display to the user's smartphone. Boring et al. [3] use the tilt technique when moving a pointer on a display using a phone and though not the same application, the execution of the technique is the same. We chose this technique because it is one handed, relatively low complexity, and much like the *Swipe* it is generally easy to use. When the direction is reversed, *Tilt* is an exact copy of the way Lucero et al. describe the technique. The *Tilt* technique is performed by *1*) pointing at a target on the large display with the phone in a stretched arm, and *2*) making a forwards tilt with the phone (fig. 4).



(a) (b) (c)

Figure 4. The *Tilt* technique is performed by doing a forward tilt motion with the phone from (a) to (c).

As mentioned, the techniques have been used in previously published papers where they were part of applications or systems, facilitating interaction between devices such as phone to display interaction and vice versa. The described techniques are different in the way they are performed but also the number of hands that are required to make them work. We chose two techniques that are one-handed (*Swipe* and *Tilt*) and two techniques that are two-handed (*Throw* and *Pinch*). Also, to vary the complexity we chose two techniques that require more steps (*Throw* and *Pinch*), and two techniques that require less steps for the user to complete the technique (*Swipe* and *Tilt*).

All the techniques make use of some combination of pointing in mid-air and touch gestures on the smartphone screen. The mid-air pointing interaction is achieved by using the Microsoft Kinect for Windows which uses a depth camera making it possible to track a user's hand in mid-air. As for the touch gestures, smartphones have an accelerometer and a touchscreen, making it possible to detect motion input and detect contact between e.g. a finger and the screen. These technologies are, in combination with each other, used to recognize the four techniques described in this section.

EXPERIMENT

The four cross-device interaction techniques mentioned above were implemented and then evaluated in a lab study in order to judge their performance compared to each other. We are interested in knowing whether or not the different techniques with different target sizes have an effect on the efficiency, accuracy, and ease of use of pushing information to a large display. Therefore, we developed an application that would allow us to run experiments and test the effect of the different techniques and target sizes. We utilized a Microsoft Kinect, a 65' inch Panasonic television with a 1920×1080 resolution and a Samsung Galaxy SII to create the experimental application. An overview of the experiment setup can be seen in Figure 5.

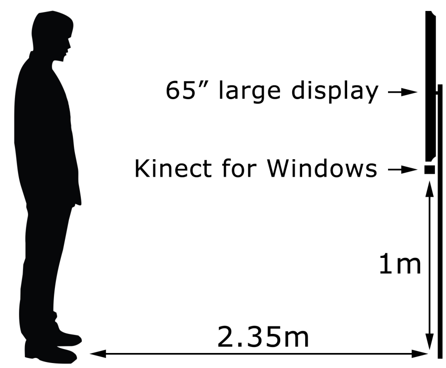


Figure 5. An overview of the entire setup of our experiment.

Implementation

The 4 techniques (*Pinch*, *Swipe*, *Throw*, and *Tilt*) were implemented in order to push data onto the large display. They were implemented in a simple and short target practice application, where the goal was to "hit" the target on the display with the shown technique.

A grid system was implemented in the test application, where each cell of the grid is a possible target. As mentioned, we were interested in measuring the effect of different target sizes on the different techniques, therefore the grid is implemented in two different sizes. The grid system can have large cells, where the grid is 5×10 cells and each cell is 61 pixels or 7.3 cm wide, or it could also have small cells, where the grid is 10×20 cells and each cell is 122 pixels, or 14.6 cm wide. The target is located in one of those cells, and scales accordingly to the size of the cell(See Figure 6a).

A red dot works as the pointer in the screen; it is the location that would be hit when the user performed the given technique. The yellow highlighted cell is the cell in which the pointer is currently located inside. This is extra feedback for the user so he knows exactly were he will hit once he performs the technique.

The developed mobile application was simple. It showed two shapes, a circle and a square, which the user could choose to push to the display. The display would tell which shape was the correct one by having that shape as the target in one of the grid cells. We chose two shapes so that it did not become a search problem with users spending too much time searching for the correct shape. We wanted the user to spend some time orienting him or her self with the phone and not just simply performing the gesture without paying any attention to the phone at all. The phone screen can be seen in Figure 6b.

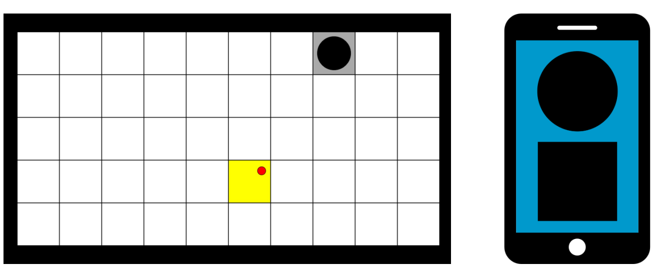
Users would control the pointer on the large display with their hands. Which ever hand was closest to the screen would determine the position of the pointer on the large screen. This meant that users could switch hands whenever they pleased at any point during the test.

The *Pinch* technique was implemented with the help of the Kinect and the touch screen on the mobile phone. This technique started by having the user pinch the shape on the screen of the mobile phone and close his or her hand around it, as if to grab it. The Kinect would then look for a opening of the hand motion, on the pointer hand, and take that as the target point.

The *Swipe* technique was implemented with the touch screen of the phone. Here, we detected when a significant swipe happened on the screen, and then use the pointer location to place the shape that was swiped up onto the screen.

The *Throw* technique was also implemented with the help of the Kinect and the accelerometer on the mobile phone. The Kinect looked at the user to recognize when a user moved the mobile phone from 10 centimeters behind the hip to 10 centimeters in front of the hip. At the same time, the phone detects when a significant change in the accelerometer happened, so to not simply detect an unintentional wave of the arm. The Kinect would then use the position of the other hand to see where on the screen the user intended to perform the *Throw* technique towards.

The *Tilt* technique was implemented mostly with the accelerometer of the phone, by checking for a significant change in the z and y axis of the accelerometer, as if tilting the phone forward.



(a) (b)

Figure 6. (a) The large display screen of the application, with the red dot in the highlighted square as the starting point and the black circle as the target. (b) The phone screen showing the two shapes.

Experimental Design

The experiment was conducted as a within-subject research, with the four different interaction techniques and two target sizes as independent variables.

The within-subject research was chosen because we wanted to minimize the amount of subjects needed in order to get a significant result. We also believed that the learning effect would not be as pronounced since the four techniques are very different from each other. We chose to investigate techniques because we were interested in learning about the way people interact with large displays, and the two target sizes to investigate precision. Sometimes users need to be as precise as possible, and sometimes they just need to be able to interact with a large display.

For the dependent variables, different measures of completion time and hit success were used, as well as a short questionnaire to get the user's satisfaction with regard to the given interaction technique. Which technique started the test was randomized in order to mitigate the learning effect on the entire set of tests. In the end, the *Pinch* gesture started 26.4% of all tests, *Swipe* started 22.7% of all tests, *Throw* started 24.5% of all tests, and *Tilt* started 26.4% of all tests. All of this was automatically logged, and every test session was also video recorded in order to be able to go through them in case we wanted to go into detail in one of the test sessions.

A simple logging mechanism was developed, which created a unique file for each user and outputted all attempts into that file. In the end, the result was a list of 53 files, one for each test participant, where each file would have a list of attempts and target size switches. Each attempt would have a time stamp, whether the user hit the target or not, whether he selected the correct shape, were the target was, and where the participant hit. These where the following measures that we were able to deduce from the log files that were generated:

Total Time: This was the time each user spent completing the test for a given interaction technique. This was measured from the time each user had hit his first target after a practice period of three tries until he had hit his last target. There were a total of 18 targets, plus the first target used for calibration.

Time per target: This was the time each user spent hitting each of the targets. This was measured as the time since each user last hit a target until he hit the next one.

Hit success: Whether or not each user hit the given target. Current pointer and target position (in both cell and pixels coordinates) were also recorded in order to give a precise measure of accuracy for each attempt in terms of *distance to target*.

Ease of use: Each user was given a questionnaire after having gone through each interaction technique. There were 6 questions, all taken from the USE questionnaire [10]. These were asked to get an understanding of how useful and easy to use each technique was. The 6 questions were the following:

- It is easy to use
- Using it is effortless
- It is easy to learn to use
- I can use it successfully every time
- I quickly became skillful with it
- I learned how to use it quickly

Users were able to rate their answers to each question on a 7 point Likert scale. We also wrote down any comments made during the experiment and combined them with the questionnaire responses to get a better understanding of the user's response to each of the techniques.

Participants

In total, 53 people took part in our experiment, which was conducted in a usability lab. The participants where between 20-45 years old (M: 24.4, SD: 4.3) and were between 1.63 and 1.95 meters tall (M: 1.82, SD:7.8). 88.7% of users were right handed, 90.6% were male, and 96.2% of them were smartphone users. Of those who owned smartphones, they had owned one for 2-15 years (M:5, SD:2.1). They were recruited through a mixture of our social network and recruitment posters around the campus.

The Experiment

Each test subject was taken into the usability lab and given a short introduction to what we were doing and why. We then explained how the system worked and what they had to do. We would hand them a phone, ask them to stand on a marked cross, so that the distance to the screen would always be the same, and start the test.

The application chose at random one of the four techniques and displayed a short explanatory movie of how to perform the technique on a screen right beside the main application display(See Figure 7a).

(a) (b)

Figure 7. (a) The main screen on the left with the tutorial video screen on the right. (b) The experiment in progress, as seen from the right.

The user would then have three practice attempts, in order to get familiar with the technique. Nothing was logged during the practice phase. A shape would appear, either a square or a circle, at one of the cells in the grid. The user would have to choose the correct shape on the phone and perform the technique with that shape selected. The shapes on the phone (fig. 6b) would randomly change positions, so that the user would have to check the phone after every technique. The target would also randomly change size from small to large or vice-versa. In reality, the target sequences where hard coded by us in such a way that there was an equal distribution of small and large targets. We also made sure that there was an equal distribution of distances between each target. We classified them as short jumps, medium jumps, and long jumps. A short jump was 2 large cells(4 small cells), a medium jump was 4 large cells(8 small cells), and a large jump was 6 large cells(12 small cells). After a practice phase of 3 practice targets, a calibration start target would be shown. This is so that we could calculate the distance between all other targets correctly. The user would then go through the rest of the test (18 targets), going through a total of 22 targets.

That means that our experiment had the following list of conditions:

- Technique (4)
- Target size (2)
- Target jump distance (3)
- Repetition (3)

This means that each user had a total of $4 \times 2 \times 3 \times 3 = 72$ targets.

After going through every target for one technique, the user would then be asked to fill in the short questionnaire regarding the technique just tried in terms of how natural it felt based on ease of use measures.

This entire process would be repeated four times in total, once for each technique. After that, we presented them with a short demographics questionnaire, in order to better understand the user. We asked them about their age, height, if they were left or right handed, if they had a smartphone, for how long, if they had any experience with a Kinect, Wii, Playstation Move, or any other similar air gesture based technologies, and how often they used them. Finally, we thanked them for their time. The entire test took on average 15 minutes.

RESULTS

We will now present the results that we achieved through out our experiment and also how they were achieved. We will first present our findings in respect to success rate(based on hit success), then in respect to efficiency(based on time per target), and finally in respect to accuracy(based on distance to target). Finally, we will look at the questionnaires and show the significant findings there in terms of ease of use. We will discuss these results later, in the Discussion section. Each of the four interaction techniques, *Pinch*, *Swipe*, *Throw* and *Tilt* were completed 18 times per participant. For each target size, each technique was performed 477 times.

Success Rate

In this section we will present the results related to the success of hitting a target. We will perform an analysis of the hit success rate and the effect of each technique with respect to target sizes.

The success rate's mean and standard deviation, M(St.D.) for each technique for small(S) and large(L) target sizes can be seen in Table 2 and in Figure 8 the results are shown as a graph.

Hit Success Means

	Pinch (n = 477)	Swipe (n = 477)	Throw (n = 477)	Tilt (n = 477)
S	0.65 (0.48)	0.91 (0.29)	0.83 (0.37)	0.58 (0.49)
L	0.78 (0.41)	0.97 (0.18)	0.94 (0.25)	0.78 (0.41)

Table 2. Mean hit and standard deviation for each technique per target for small(S) and large(L) targets.

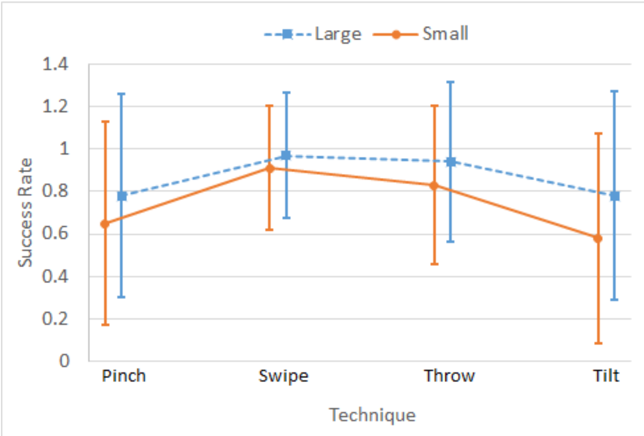


Figure 8. Mean and standard deviation for each technique in regards to hit success rate per target.

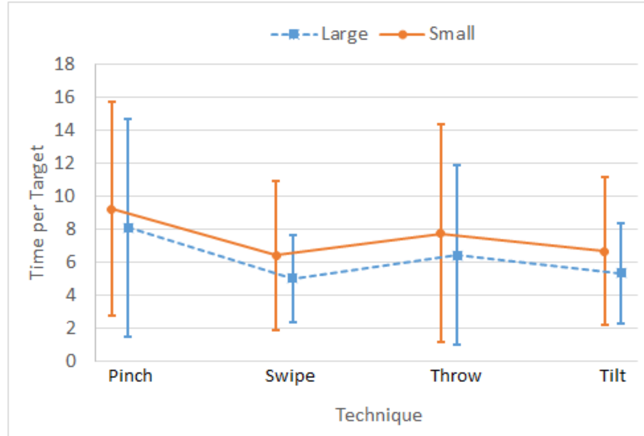


Figure 9. Mean and standard deviation for each technique in regards to time per target.

In order to see the effect of each technique on the hit ratio per target for the different target sizes, we performed two different one-way ANOVA's, where we split the data between the two different target sizes. We then performed a post-hoc pairwise LSD test to see where the significant difference were.

For the the small target, we got ($p < 0.001$), ($F(2.722, 1295.674) = 62.754$), (Greenhouse-Geisser correction: 0.907). The pairwise test showed that all techniques were significantly different. *Pinch* and *Swipe* had ($p < 0.001$), *Pinch* and *Throw* ($p < 0.001$), *Pinch* and *Tilt* ($p = 0.031$), *Swipe* and *Throw* ($p = 0.001$), *Swipe* and *Tilt* ($p < 0.001$) and finally, *Throw* and *Tilt* had ($p < 0.001$).

For the large target, we got ($p < 0.001$), ($F(2.472, 1176.749) = 42.773$), (Greenhouse-Geisser correction: 0.824). The pairwise test showed that all of the techniques, with the exception of *Pinch* and *Tilt*, were statistically different from each other. The results were as following: *Pinch* and *Swipe* ($p < 0.001$), *Pinch* and *Throw* ($p < 0.001$), *Pinch* and *Tilt* ($p = 1.000$), *Swipe* and *Throw* ($p = 0.025$), *Swipe* and *Tilt* ($p < 0.001$), and finally *Throw* and *Tilt* ($p < 0.001$).

Efficiency

In this section we present the efficiency results which defines the amount of time spent performing a technique. We perform an analysis of the efficiency and the effect of each technique with respect to target sizes.

Table 3 shows the mean time per target in seconds for each of the techniques as well as their standard deviation for both target sizes.

Time per Target Means

	Pinch (n = 477)	Swipe (n = 477)	Throw (n = 477)	Tilt (n = 477)
S	9.23 (6.48)	6.41 (4.49)	7.73 (6.60)	6.67 (4.49)
L	8.09 (6.60)	5.01 (2.66)	6.42 (5.43)	5.33 (3.04)

Table 3. Mean time and standard deviation for each technique per target for small(S) and large(L) targets.

To get the effect of each technique on the time for the different target sizes, we performed two one-way ANOVA's, one for the small target and another for the large target. A post-hoc pairwise LSD test to see where the significant difference were.

For the the small target, we got ($p < 0.001$), ($F(2.740, 1304.290) = 26.523$), (Greenhouse-Geisser correction: 0.913). The pairwise test showed that all techniques were significantly different except *Swipe* and *Tilt*. *Pinch* and *Swipe* had ($p < 0.001$), *Pinch* and *Throw* ($p < 0.001$), *Pinch* and *Tilt* ($p < 0.001$), *Swipe* and *Throw* ($p < 0.001$), *Swipe* and *Tilt* ($p = 0.354$) and finally, *Throw* and *Tilt* had ($p = 0.004$).

For the large target, we got ($p < 0.001$), ($F(2.221, 1057.144) = 44.539$), (Greenhouse-Geisser correction: 0.740). The pairwise test showed that all of the techniques, with the exception of *Swipe* and *Tilt*, were statistically different from each other. The results were as following: *Pinch* and *Swipe* ($p < 0.001$), *Pinch* and *Throw* ($p < 0.001$), *Pinch* and *Tilt* ($p < 0.001$), *Swipe* and *Throw* ($p < 0.001$), *Swipe* and *Tilt* ($p = 0.077$), and finally *Throw* and *Tilt* ($p < 0.001$).

Accuracy

Finally, we will perform an analysis of the accuracy and the effect of each technique with respect to target sizes. Here, we took three different measures of accuracy; the distance between where the user hit and the target cell as well as taking the x and y axis independently. These were all measured in pixels. This was because there were signs that certain techniques might miss in a specific direction, and we wanted to see if the data supported that. An overview of the distance mean and standard deviation in pixels can be seen in Table 4 and Figure 10.

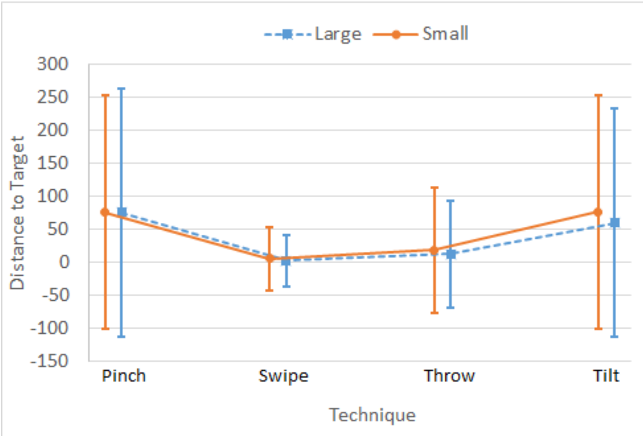


Figure 10. Mean and standard deviation for each technique in regards to distance.

Distance Means

	Pinch (n = 477)	Swipe (n = 477)	Throw (n = 477)	Tilt (n = 477)
S	75.97 (176.15)	5.40 (47.81)	18.60 (95.29)	76.51 (177.45)
L	75.41 (187.58)	2.29 (38.82)	12.37 (81.36)	59.88 (172.22)

Table 4. Mean and standard deviation for the distance for each technique per target for small(S) and large(L) targets.

We performed two one way ANOVA's, one for each target size, on the distance data.

For the the small target, we got ($p < 0.001$), ($F(2.341, 1114.249) = 37.504$), (Greenhouse-Geisser correction: 0.780). The pairwise test showed that all techniques were significantly different except *Pinch* and *Tilt*. *Pinch* and *Swipe* had ($p < 0.001$), *Pinch* and *Throw* ($p < 0.001$), *Pinch* and *Tilt* ($p = 0.961$), *Swipe* and *Throw* ($p = 0.008$), *Swipe* and *Tilt* ($p < 0.001$) and finally, *Throw* and *Tilt* had ($p < 0.001$).

For the large target, we got ($p < 0.001$), ($F(2.176, 1036.004) = 33.315$), (Greenhouse-Geisser correction: 0.725). The pairwise test showed that all of the techniques, with the exception of *Pinch* and *Tilt*, were statistically different from each other. The results were as following: *Pinch* and *Swipe* had ($p < 0.001$), *Pinch* and *Throw* ($p < 0.001$), *Pinch* and *Tilt* ($p = 0.171$), *Swipe* and *Throw* ($p = 0.015$), *Swipe* and *Tilt* ($p < 0.001$) and finally, *Throw* and *Tilt* had ($p < 0.001$).

X and Y Distance Means

	Pinch (n = 477)	Swipe (n = 477)	Throw (n = 477)	Tilt (n = 477)
S XD.	54.33 (140.87)	3.78 (40.15)	10.32 (81.84)	49.23 (151.36)
L XD.	55.32 (159.88)	1.88 (31.74)	10.32 (74.01)	49.23 (157.35)
S YD.	42.90 (110.30)	2.00 (26.16)	8.00 (49.37)	39.3 (99.82)3
L YD.	37.41 (104.18)	1.14 (22.37)	4.33 (34.21)	22.86 (73.75)

Table 5. Mean and standard deviation for distance on the X-Axis(XD) and distance on the Y-Axis(YD) for each technique per target for small(S) and large(L) targets.

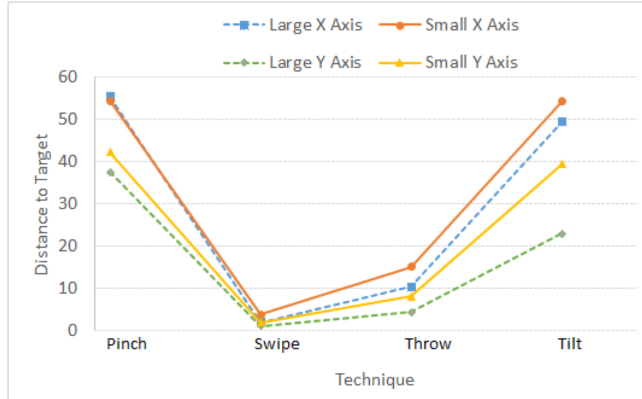


Figure 11. Mean and standard deviation for each technique in regards distance on the x-axis and y-axis.

To get a better understanding of the distance we decided to examine the distance in regards to the x-axis and in regards to the y-axis. The results can be seen in Table 5 and Figure 11

Results of Ease of Use Questionnaire

Our questionnaire was based on USE, which used Likert scale 4.2, when encoding the data we did it as continuous variable, as such *"strongly disagree"* got a value of 1, and *"strongly agree"* a value ot 7. After that the cumulative value per technique, based on the different questions, was calculated, the data was ploted and presented in Figure 12.

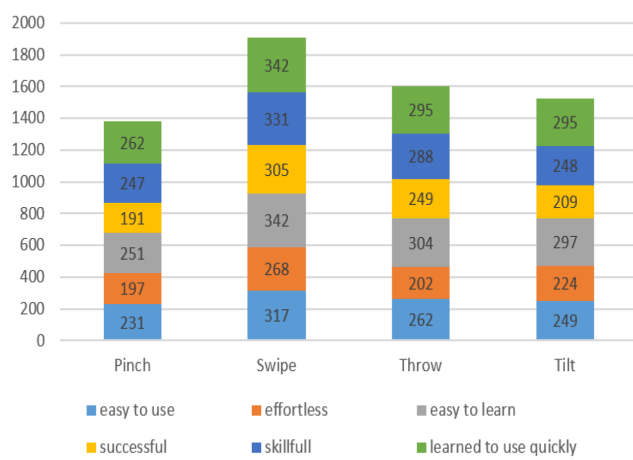


Figure 12. Cumulative values of survey questions per technique

A One-Way MANOVA was applied ($F(18, 574.66) = 5.118$, $p < 0.000$) which showed that there is statistical differences between each techniques. In order to specify where this differences lie we perform an post hoc test, we summarize the results in regards to each question from the survey, as follow: *1)* "It is easy to use" *Swipe* is statistically different, however there is no statistical difference between *Throw*, *Tilt*, and *Pinch*; *2)* "Using it is effortless" *Swipe* is statistically different, however there is no statistical difference between *Throw*, *Tilt*, and *Pinch*; *3)* "It is easy to learn to use" *Swipe* and *Pinch* are statistically different, however there is no statistical difference between *Tilt*, and *Throw*; *4)* "I can use it successfully every time" *Swipe* and *Throw* are statistically different, there is no statistical difference between *Tilt*, and *Pinch*; *5*) "I quickly became skillful with it" *Swipe* and *Throw* are statistically different, however there is no statistical difference between *Pinch*, and *Tilt*; *6*) "I learned how to use it quickly" *Swipe* is statistically different, however there is no statistical difference between *Throw*, *Tilt*, and *Pinch*. The results show that the ratings received by the techniques differ in the different aspects of the areas covered by the survey.

DISCUSSION

When discussing our results, it is important to redefine the terms that we use. When talking about success rate, we mean whether or not the technique hit the given target. When we mention efficiency, we are talking about the time it takes to successfully perform that technique. When talking about accuracy, we mean the distance the attempt was from the target(in pixels). It is also important to note that the standard deviation in some of our measures are quite high. This is primarily because the experimental system was not robust enough to get a perfect reading on the users intention to perform a technique. Sometimes, the system would misunderstand the gesture a user made, so that it either activated too early, or did not activate all. Activation means that the system interpreted the gestures the user performed as an attempt to hit the target. This is further discussed in Section 6.7

When looking at the results on the effect of the four techniques, *Pinch*, *Swipe*, *Throw* and *Tilt*, as well as the two target sizes, small and large, on the time per target, the results tell a rather interesting story.

Success Rate

If we look at the results in regards to the effect of the technique on the success rate of each attempt, it is interesting to note that the two techniques that were not significantly different from each other were *Tilt* and *Pinch*. These two techniques both used the hand that controlled the pointer to activate the technique. When tilting the phone forward, usually the hand would move together with the phone causing the pointer to displace itself from the users intended position. When releasing the *Pinch*, the Kinect would sometimes reevaluate the location of the hand joint, now that it could see the entire hand, which would also cause the pointer to displace itself from the intended position. *Pinch* and *Tilt* were also the techniques that had the largest amount of activation errors due to the implementation of the system. Sometimes, users would show large amount of their palms to the Kinect during a *Pinch*, even though their hand was closed, causing the Kinect to interpret that as an opening of the hand and activate the technique. *Tilt* would sometimes activate if the user moved the mobile around too quickly, especially when orienting the pointer up and down on the screen.

Swipe and *Throw* both had reasonably high success rates. *Throw* did not require the user to actually move the pointer hand while activating the technique. While *Swipe* did require the user to perform some movement on the hand that was used as a pointer, it was very little movement. This is also a technique all smartphone users are familiar with, since a lot of applications use some form swiping to activate some functionality.

Efficiency

There was a significant difference between all techniques, with the exception of *Swipe* and *Tilt*. These were the two one handed techniques that we chose. The range of movement needed in order to activate these two techniques was rather limited, the full motion could be achieved quite quickly and is quite similar for both of them. This is why they are not statistically different from each other. *Swipe* and *Tilt*, are on average, at least a second faster then the other two techniques. Their standard deviations are also smaller, which means that users were more consistent, with regards to how long it took to hit each target, with these two techniques.

Looking at the two other techniques, *Pinch* and *Throw*, their times also reflect the range of motion needed in order to activate each technique. *Pinch* requires the user to pinch the shape on their phone, lift their hand up, direct it on the screen, and then finally let go. This can be seen in its mean, where it takes almost 1.59 seconds longer to perform than the second longest technique, *Throw*. *Throw* also requires a considerable range of motion in order to activate: point with one arm, select the shape on the phone with the other arm, bring your arm back and then finally swing it forward. Both two handed techniques take significantly longer to perform than their one handed counter-parts.

We noticed that users would spent relatively little time getting into the general vicinity of the target, and would spend most of their time per attempt getting the pointer on top of the actual target. This was more pronounced for the small target, were users would perform smaller, more careful adjustments in order to not overshot the target, which can be seen when comparing the mean times of each technique for the two different target sizes.

Accuracy

If we look at the results regarding the distance from the target, it paints quite a clear image of which of the techniques are more accurate. *Swipe* is by far the most accurate of the four techniques. It is so precise, that it is actually more precise for the small target than all other techniques are for the large target. Figure 13 is an image that shows the location of each hit compared to the given target cell.

Figure 13 shows that *Swipe* and *Throw* have a large concentration of hits inside the target cell, where *Pinch* and *Tilt* have quite a spread of hits outside the target cell. This shows us that users are capable of hitting the target with *Swipe* and *Throw* more consistently and accurately than with the other two techniques by considerable amounts.

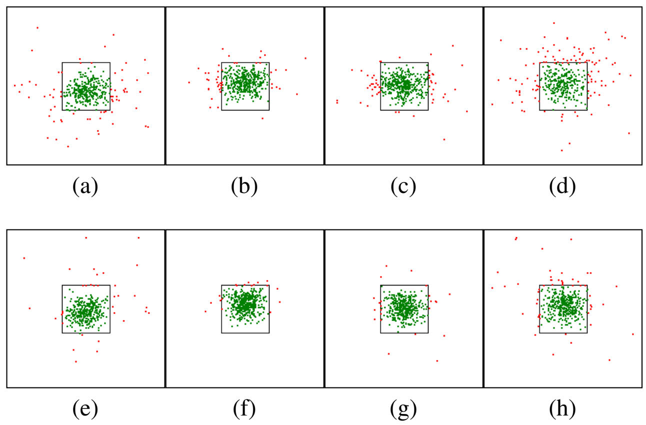


Figure 13. Hit location illustration for each technique. Green is a hit, red is a miss (a) Pinch (Small target) (b) Swipe (Small target) (c) Throw (Small target) (d) Tilt (Small target) (e) Pinch (Large target) (f) Swipe (Large target) (g) Throw (Large target) (h) Tilt (Large target)

Also, an interesting thing to notice is that there is a trend where the largest distance from the target is located on the x axis. This can be seen in Table 5, where the distance is broken into the two different axis. For all four techniques, the x axis has a larger mean distance from the target then the y axis. Further research should be done though if one were to make some conclusive statement in regards to the effect of the different axis on the accuracy of the techniques.

Target Size

If we compare the effect of the target size on the different techniques, it does not have that big of an impact on the success rate of each technique compared to each other. *Swipe* and *Throw* are still the techniques with the highest success rate. Our results also show that target size does not change the efficiency of each technique compared to each other. The same is true for the distance from the target cell. Our results point towards a tendency were the size of the target does not influence the measured effectiveness of each technique relative to each other. This of course would require further research, since a sample size of two target sizes is not enough.

Easy of Use

Looking at the questionnaire, we can look at each question and see there is a trend. If the user gave it a high score, then he strongly agreed with the given question. We can then say the accumulated scores of each technique for all questions show a tendency towards the user agreeing that the given technique was easy to use and hence more natural to use. The higher the score, the easier the user felt the given technique was to perform. This means that in general, users considered *Swipe* much easier to use then the other techniques. *Throw* and *Tilt* were considerably close to each other, while *Pinch* trailed quite significantly behind. It is interesting to note that *Throw* and *Tilt* were so close to each other in ease of use, even though *Throw* outperformed *Tilt* considerably, both in regards to time and hit success, as well as the consistency of the technique, shown by the standard deviation.

User Comments

There is also a qualitative aspect to take into account here, which is not reflected well in the surveys or the test results, but were recorded on video and notes during each experiment. For example, a large finding was that correct mapping between the direction in which the user is pointing and the pointer icon on the screen is critical to the success of the application. It is extremely important for the experience of the user to have as close to absolute mapping as possible. Eleven users mentioned having trouble reaching all areas of the screen, but almost all users showed sign of trouble, by for example standing on their toes or stretching their arms as far as possible. One user got so frustrated that she asked for a chair to stand on. With more or different sensors, placed on hands, fingers and phones, we could have had a more precise pointer by being able to determine the direction of the phone and the arm and not solely rely on the position of the users hands. This would most likely lead to more precise results, because the the mapping between the pointer and the users pointing direction would be much closer to a absolute mapping.

In regards to the mobile phone, four users complained that the screen was too small when performing a *Pinch*, making it hard to precisely select the correct shape. Four other users complained that the screen was too large when performing a swipe, since it was hard to reach the correct shape with their thumbs while still maintaining precision with the pointer. Four users mentioned that it was hard to orient themselves with the phone while performing the *Throw* technique, having to break their flow to look down on the phone to select the correct shape. Three users mentioned the same problem with *Swipe*, whenever the targets were too high. This sometimes lead to the mobile phone covering up the target and making it hard for the user to orient themselves to the large display. This was an effect of the relative mapping though, since it was hard to see the screen when their arm was stretched far above their head in order to reach the high targets. The same error could have occurred with *Tilt*, since the user might also end up lifting the phone in front of their field of view, but none of the users mentioned it there.

There is also a the learning aspect of each technique. Six people actually mentioned that the *Pinch* technique was hard to learn, and a large portion of the participants had to be told more specifically how to perform it. The same held true for the *Throw* technique, a large number of the participants had to be told that they had to perform a slightly larger motion in order for the application to understand that a throw motion was being attempted. Very few of the participants had to have further instructions on how the *Swipe* technique worked, and few people needed further help with the *Tilt*. This is most likely a combination of the complexity of some of the techniques as well as the tutorial movies not being descriptive enough. This also lead to frustration, were users thought they were performing the technique correctly and nothing was happening. Four users mentioned being frustrated by the *Pinch* technique, while three users got frustrated with the *Tilt* technique

The fatigue effect is also something to take into consideration. 13 people mentioned being fatigued through out the test and 11 of them first mentioned it during the *Throw* technique. Some users commented that it was because one arm had nothing to do but be uplifted and point to the screen, while the other arm performed all the motion. One user mentioned it would not have been that noticeable if the pointing arm had some motion to perform. We have hopefully accounted for this by rotating the order in which the techniques are performed.

Finally, there is also the fun aspect to take into consideration. Nine users actually mentioned having fun while performing the *Pinch* technique. They compared it to casting a spell or causing explosions on the screen. Three other users mentioned that this technique was especially interesting. It is still worth remembering though that *Pinch* was, by large, the hardest technique for users to learn.

Limitations

There are of course some limitations to the system we developed. Firstly, the intention with the *Tilt* technique was that the users would point and tilt with the phone, but because of our implementation, it was possible for users to point with one hand and tilt the phone with the other. The same holds true for the *Swipe* technique, where users were able to point with one hand and swipe with the other.

The way the system detects an open hand is not very robust: sometimes, depending on the profile of the hand, it misreads the users intentions and believes the user opened his hand.

The system also has a very narrow definition of what throwing means. This is something that can be seen when users were told to "throw" the data from the phone to the screen. Some would perform a much larger tilt motion, others would perform a baseball-like pitching motion.

The Kinect also had some problems determining where the different arm joints were. If the elbow joint was directly behind the hand joint from the Kinects perspective, it would cause the pointer to move erratically since the Kinect was not absolutely sure were the hand joint was. Another problem occurred when the user put their two hands close to each other. The Kinect would have problems determining where the individual hand joints were located.

CONCLUSION

We conducted a study on cross-device interaction techniques, where our focus was on pushing data from a mobile device onto a large display. We compared four different techniques (*Pinch*, *Swipe*, *Throw* and *Tilt*) with two different target sizes. Our main concern was investigating the success rate, efficiency, accuracy and ease of use of the user while using the different interaction techniques.

Our findings show that *Swipe* was the technique with the highest success rate, and at the same time, the most efficient and accurate technique. This was also the technique the users felt easiest to use. We also found that *Pinch* was considered to be a fun and entertaining technique by many of our users. Finally, we also found that the mapping between the screen and the users pointing direction is critical to the applications success. Great care should be taken to achieve as close to an absolute mapping as possible.

FUTURE RESEARCH

Based on our experiment, in which we used two target sizes, we can not determine the difference in effect between targets of different sizes on a large display. As such we would suggest more research including a greater varity of target sizes.

Whereas our experiment only consider push techniques, we would strongly suggest looking at pull techniques for cross-device natural user interaction with large displays. With pull techniques we imagine research investigating the opposite direction i.e. pulling information from a large display to a mobile device. The techniques which may be preferred for pushing information to a large display might not be the best choice for pulling information.

Our experiment has exclusively been concerned with specific measures like success rate, efficiency and accuracy for each technique, and we could suggest for future research that other measures be included in experiment. This includes, but is not limited to, measures on user experience and which techniques users prefer to use for interaction with displays located in public places.

We focused only on the interaction between mobile phones and large displays. In the future though, the range of different personal devices will probably be much more widespread than today, and we suggest further research in this area. An example of this research is the interaction between large displays and devices such as smart watches, tables, and smart glasses.

As a final suggestion, a framework for cross-device natural user interactions might help, for example, developers and researchers with techniques, guidelines and designs for interacting with large displays in the future.

ACKNOWLEDGMENTS

We would like to thank Jeni Paay for the excellent guidance we received throughout the entire project. We would like to thank Dimitrios Raptis for helping us set up the experiment as well as understanding all the data we got out of it. We would also like to thanks Jesper Kjeldskov for making sure we had all the resources we needed. Finally, we would like to thank all the test participants in our experiment.

REFERENCES

1. Hrvoje Benko and Andrew D. Wilson. 2010. Multi-point Interactions with Immersive Omnidirectional Visualizations in a Dome. In *ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*. ACM, New York, NY, USA, 19–28. DOI: http://dx.doi.org/10.1145/1936652.1936657

2. Sebastian Boring, Manuela Altendorfer, Gregor Broll, Otmar Hilliges, and Andreas Butz. 2007. Shoot & Copy: Phonecam-based Information Transfer from Public Displays Onto Mobile Phones. In *Proceedings of the 4th International Conference on Mobile Technology, Applications, and Systems and the 1st International Symposium on Computer Human Interaction in Mobile Technology (Mobility '07)*. ACM, New York, NY, USA, 24–31. DOI: http://dx.doi.org/10.1145/1378063.1378068

3. Sebastian Boring, Marko Jurmu, and Andreas Butz. 2009. Scroll, Tilt or Move It: Using Mobile Phones to Continuously Control Pointers on Large Public Displays. In *Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7 (OZCHI '09)*. ACM, New York, NY, USA, 161–168. DOI: http://dx.doi.org/10.1145/1738826.1738853
4. Andrew Bragdon, Rob DeLine, Ken Hinckley, and Meredith Ringel Morris. 2011. Code Space: Touch + Air Gesture Hybrid Interactions for Supporting Developer Meetings. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11)*. ACM, New York, NY, USA, 212–221. DOI: http://dx.doi.org/10.1145/2076354.2076393
5. Xiang 'Anthony' Chen, Tovi Grossman, Daniel J. Wigdor, and George Fitzmaurice. 2014. Duet: Exploring Joint Interactions on a Smart Phone and a Smart Watch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 159–168. DOI: http://dx.doi.org/10.1145/2556288.2556955
6. Larry Francis. 1997. Mobile Computing: A Fact in Your Future. In *Proceedings of the 15th Annual International Conference on Computer Documentation (SIGDOC '97)*. ACM, New York, NY, USA, 63–67. DOI: http://dx.doi.org/10.1145/263367.263378
7. Peter Hamilton and Daniel J. Wigdor. 2014. Conductor: Enabling and Understanding Cross-device Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2773–2782. DOI: http://dx.doi.org/10.1145/2556288.2557170
8. Kaori Ikematsu and Itiro Siio. 2015. Memory Stones: An Intuitive Information Transfer Technique Between Multi-touch Computers. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications (HotMobile '15)*. ACM, New York, NY, USA, 3–8. DOI: http://dx.doi.org/10.1145/2699343.2699352
9. Andrés Lucero, Jussi Holopainen, and Tero Jokela. 2012. MobiComics: Collaborative Use of Mobile Phones and Large Displays for Public Expression. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '12)*. ACM, New York, NY, USA, 383–392. DOI:http://dx.doi.org/10.1145/2371574.2371634
10. AM Lund. 2001. Measuring Usability with the USE Questionnaire. STC Usability SIG Newsletter. (2001).
11. Nicolai Marquardt. 2011. Proxemic Interactions in Ubiquitous Computing Ecologies. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, New York, NY, USA, 1033–1036. DOI: http://dx.doi.org/10.1145/1979742.1979691
12. Nicolai Marquardt, Ken Hinckley, and Saul Greenberg. 2012. Cross-device Interaction via Micro-mobility and F-formations. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, USA, 13–22. DOI: http://dx.doi.org/10.1145/2380116.2380121
13. Brad A. Myers. 2001. Using Handhelds and PCs Together. *Commun. ACM* 44, 11 (Nov. 2001), 34–41. DOI:http://dx.doi.org/10.1145/384150.384159
14. Heidi Selmer Nielsen, Marius Pallisgaard Olsen, Mikael B. Skov, and Jesper Kjeldskov. 2014. JuxtaPinch: Exploring Multi-device Interaction in Collocated Photo Sharing. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services (MobileHCI '14)*. ACM, New York, NY, USA, 183–192. DOI: http://dx.doi.org/10.1145/2628363.2628369
15. Jun Rekimoto. 1997. Pick-and-drop: A Direct Manipulation Technique for Multiple Computer Environments. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology (UIST '97)*. ACM, New York, NY, USA, 31–39. DOI: http://dx.doi.org/10.1145/263407.263505
16. Jun Rekimoto. 1998. Multiple-Computer User Interfaces: A Cooperative Environment Consisting of Multiple Digital Devices. In *Proceedings of the First International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture (CoBuild '98)*. Springer-Verlag, London, UK, UK, 33–40. http://dl.acm.org/citation.cfm?id=645968.674728
17. Jürgen Scheible, Timo Ojala, and Paul Coulton. 2008. MobiToss: A Novel Gesture Based Interface for Creating and Sharing Mobile Multimedia Art on Large Public Displays. In *Proceedings of the 16th ACM International Conference on Multimedia (MM '08)*. ACM, New York, NY, USA, 957–960. DOI: http://dx.doi.org/10.1145/1459359.1459532
18. Dominik Schmidt, Julian Seifert, Enrico Rukzio, and Hans Gellersen. 2012. A Cross-device Interaction Style for Mobiles and Surfaces. In *Proceedings of the Designing Interactive Systems Conference (DIS '12)*. ACM, New York, NY, USA, 318–327. DOI: http://dx.doi.org/10.1145/2317956.2318005
19. Julian Seifert, Adalberto Simeone, Dominik Schmidt, Paul Holleis, Christian Reinartz, Matthias Wagner, Hans Gellersen, and Enrico Rukzio. 2012. MobiSurf: Improving Co-located Collaboration Through Integrating Mobile Devices and Interactive Surfaces. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces (ITS '12)*. ACM, New York, NY, USA, 51–60. DOI: http://dx.doi.org/10.1145/2396636.2396644

20. Mikael B. Skov, Jesper Kjeldskov, Jeni Paay, Heidi P. Jensen, and Marius P. Olsen. 2015. Investigating Cross-Device Interaction Techniques: A Case of Card Playing on Handhelds and Tablets. In *Proceedings of the Annual Meeting of the Australian Special Interest Group for Computer Human Interaction (OzCHI '15)*. ACM, New York, NY, USA, 446–454. DOI: http://dx.doi.org/10.1145/2838739.2838763

21. Robert Walter, Gilles Bailly, Nina Valkanova, and Jörg Müller. 2014. Cuenesics: Using Mid-air Gestures to Select Items on Interactive Public Displays. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services (MobileHCI '14)*. ACM, New York, NY, USA, 299–308. DOI: http://dx.doi.org/10.1145/2628363.2628368