MASTER 2 RESEARCH - INTERACTION SPECIALTY

Virtual navigation on wall-sized displays

Author:
Bruno FRUCHARD

Supervisors:
Olivier CHAPUIS
Emmanuel PIETRIGA

Hosting lab/enterprise:
Inria Saclay - Ile-de-France - LRI
Team ILDA

April 1st 2015 – September 30th 2015

Secrétariat - tel: 01 69 15 66 36 Fax: 01 69 15 42 72
email: Murielle.Benard@u-psud.fr

Contents

Contents **i**

1 Introduction **1**

2 Related Work **3**

3 Virtual navigation **6**
3.1 Technologies 6
3.2 Interaction techniques 8

4 User study **14**
4.1 Tasks 14
4.2 Experiment 16

5 Conclusion and perspectives **24**

Bibliography **25**

A Questionnaire **28**

Summary

We present in this report the work we made for an internship about virtual navigation on wall-sized displays. Over the years, scientists built a lot of systems to interact with computers involving different parts of the human body. Yet, wall-sized displays were created not so long ago. So some questions remain unanswered regarding the way to interact with them. Depending on the context, one may use conventional mouse and keyboard as well as touching the display directly, but a bad choice could lead to more time consumed or fatigue felt. For this internship we focused on the context of virtual navigation (pan and zoom interaction). We compared two types of interaction on it : direct interaction using touch on the wall, and indirect interaction using a mobile device to interact remotely. We wanted to investigate the advantages and drawbacks both could have and chose 5 interaction techniques picked from the literature to use them as a comparison factor. We ran an experiment using a simple navigation task using both types of interaction. Our results showed that using indirect interaction for navigation was faster than direct interaction, apparently due to the lack of overview on the display when interacting close to the wall. Moreover, we observed that some techniques performed better than others, which is an opportunity to focus on them more carefully in future work.

Keywords

Human-Computer Interaction, Wall-sized display, Virtual navigation, Direct interaction, Indirect interaction, Pan & Zoom

CHAPTER 1

Introduction

The quantity of data stored by computers all around the world keeps growing every day and there is a growing need to visualize it. Researchers and designers have proposed many techniques and visualizations to handle that quantity of information, but as it continues to grow, inventing interaction techniques on computers will reach a limit. Scientists wanted to investigate the idea of jumping to the next scale which led them to build wall-sized displays. They are made of several screens working in a single shared information space. Their size may vary, but in the case of this internship, we used the WILDER platform set up at Inria. It is composed of 10 computers controlling 7 to 8 screens each, measuring approximately 5x2 meters. The reasons why we built these displays are that human vision let us see approximately everything in a 180 ° radius (with coarse information on the sides) in front of us. Yet computer screens fill less than 40% of that field. Thus, increasing the size of the display takes better advantage of human vision. Moreover, it is an opportunity to investigate new types of interaction using the whole human body. Since we can stand in front of a wall, it is possible to use our legs and arms to interact, closing the gap between everyday life interactions and interacting with computers. Finally, the high-resolution enabled by such displays gives an importance to the position of each user interacting with it. From afar, one can appreciate an overview of the information displayed as well as getting more precise information when standing close to the display.

In spite of a large quantity of information displayed, we still need to navigate through data; the quantity of data will always grow faster than display size. Interacting with this kind of display is challenging since it is inconvenient to use classical ways of interacting such as keyboards or mice; by using these, we lose the capacity to physically move in front of the wall. It has led researchers to look for new techniques to adapt to these new constraints. During this internship we focused on the kind of techniques as well as the type of interaction we could use on a wall. Our goal was to compare tactile interaction on the wall (called **direct** interaction) versus distant interaction (called **indirect** interaction) in the context of tasks involving virtual navigation. To be sure we did not favor one type of interaction over the other, we used the same techniques for both conditions, and used tactile surfaces in both cases. For direct interaction, all techniques were performed by interacting directly on the wall and for indirect interaction we used a tablet which transmitted all interactions through wi-fi. Virtual navigation typically involves the use of pan and zoom interaction, so we wanted to handle both all along this internship. Yet, after a while we understood this would lead to complex set of studies since few works had already been done on wall-sized displays concerning virtual navigation. We thus decided to focus only on pan techniques and had to build a task representing an actual panning task, from which we could draw interesting observations about direct and indirect interaction. In order to achieve that, we went through an iterative process to define an experiment.

The structure of this report is as follows. In the first part, we look at the technologies we were able to use with the wall and their purpose, followed by a detailed description of each technique we wanted to evaluate, as well as their advantages and drawbacks. In the second part, we describe the user study we conducted and report our observations. To conclude, we talk about the perspectives of future work following the trail of the results we found.

Chapter 2

Related Work

Scientists working in the HCI field have studied pointing tasks for many years, resulting in the creation of explicit laws such as Fitts' law [8]. They found that these laws may apply in a certain context, for example using mice and keyboards, therefore giving the possibility to evaluate an interface. Knowing that, some researchers have been trying to evaluate pointing tasks on a wall-sized display in order to validate known laws in that context, or even finding new ones. Using a wall-sized display, one has the possibility to use either absolute pointing or relative pointing. The first uses user input on the wall without modifying, whereas the second modifies it relative to specific rules (usually resulting in a movement acceleration). Forlines et al. proposed *Hybrid Pointing* [9] a pen-based interface that lets users switch between absolute and relative pointing easily. They found, as hypothesized, that users tend to use absolute pointing for nearby targets and relative pointing for distant targets. The idea of switching from one type of control to the other seems a good idea, yet in the case of having more than one target to select, the problem may reside in the lack of overview while standing too close to the display.
Thus, some works have focused on pointing at targets on a wall-sized display from a distance. It may be achieve using laser pointers but does not seem to be the best idea looking at Myers et al.'s work [18]. We could also use mobile devices, but in that case, absolute pointing is hard to perform because of the differences in displays' size. The movement will then be really slow and only useful for tiny distances. Nancel et al. [19] proposed an acceleration function mapping the movement speed to the distance travelled, as well as tools for relative pointing using either a head-tracking system or the screens' size composing the display. These techniques perform well, but other ways of interacting may be investigated as mid-air gestures. We may refer to *Pointable*, a technique working on tabletop displays using mid-air gestures by Banerjee et al. [4], which could be easily adapted to wall-sized displays. But as showed by Nancel et al. [20], the use of mid-air gestures in front of a wall may lack some physical feedback and results in deteriorated performances.

Pointing tasks are an interesting subject to investigate but are not the only possible interaction on a wall-sized display. The size of such displays lets users move freely in front of them, making physical navigation part of the interaction process. Yet some works have been trying to build interactions from a desktop to this new type of display. That is what Malik et al. [16] did with a vision-tracked multi-finger gestural input proposing bi-manual interaction. The way they built this bi-manual tool is clever but the idea of sitting at a desktop to interact with a wall-display does not take advantage of the space in front of the display. The importance of the physical space may even be an advantage in the time taken to perform a task as showed by Ball et al. [3]. As proposed with the *Shadow reaching* technique by Shoemaker et al. [24], users' position could even be a way of interacting, proposing new perspectives of interaction. Nevertheless, the opportunity for collaboration or innovative techniques seems diminished in this case. Other systems such as *Screenfinity* developed by Schmidt et al. [23] use user position to adapt content displayed. It gives importance to user position but adds implicit interaction in front of a wall which can lead to problems such as "Midas touch" [11][1] or disturbances while multiple users are interacting. The same idea has been investigated with *MirrorTouch* developed by Müller et al. [17], by adding a touch interface to the display. Their work showed that once users have chosen a modality to interact, they do not switch to the other one. In the same trail, Jakobsen et al. [12] did investigate trade-offs using direct or indirect interaction on a wall in order to detect any effect of the user position on the task performed. They forced users to step back from the display and showed that users do no tend to interact with touch again after that but keep interacting from a distance. Yet, their tasks only focus on pointing, without having to navigate virtually.

One advantage of wall-sized displays is the quantity of data we are able to visualize on them. As we saw, it does not remove the need to navigate through these data. Virtual navigation on computers is widespread now, usually while interacting on virtual maps, and uses common techniques as pan and zoom. Since interacting with a mouse and keyboard does not seem to be a valid option with a wall-sized display, tactile techniques have been investigated on walls. One interesting technique is *CycloStar*, proposed by Malacria et al. [15], that enables clutch-free movement. It uses elliptical movements to switch easily from pan to zoom gestures without releasing the finger from the display. Interacting directly on a wall is really tiring because the user has to hold his arm up; this is known as the "gorilla-arm" effect. Clutch-free techniques avoid useless movements but do not answer this problem completely.
Since virtual navigation has not been investigated much on wall-sized display, techniques adapted specifically for this are rare. Yet they could be associated with a tactile surface that eases the adaptation of known tactile techniques. We may refer to *RubberEdge* described by Casiez et al. [5] that uses a trackpad to switch from absolute movement to relative movement using a physical area as feedback. The idea behind that technique is for example usable in a wall context, but would need to be adapted. Other techniques such as *Flick-and-brake* proposed by Baglioni et al. [2] extend the flick technique, which allows quick throwing gestures to give a speed to a panning movement with a dampening effect, to a non-dampening flick.

[1] interacting without willing it

The interest we find with that technique in a wall context is to avoid clutching and only stop the movement when needed. Typical zooming techniques as pinch-to-zoom (that uses the gap between two fingers to set a zoom factor) are usable on a wall but require two hands to perform due to the display size. Techniques like *DragMag* proposed by Ware et al. [26] are interesting to avoid losing an overview of the data displayed. Several *DragMag*s can be displayed simultaneously, which can provide effective support to collaborative work.

Finally, the advantages and drawbacks of wall-sized displays are sometime not that clear. Liu et al. [14] tried to understand better the trade-off between either a computer or a wall. They built a classification task based on their experience in wall-sized display and showed that the more information one has to deal with, the more time it will take using a computer. We may also refer to Andrews et al. [1]'s work about clusters of screens to deal with analytic tasks. They did not use a wall-sized display but a smaller cluster of screens. Nevertheless the idea of using a physical space to classify the information seems efficient to perform cognitive tasks. The most important advantage and drawback of wall-sized displays is their size. We saw many arguments showing why this is an advantage but it may become a drawback when talking about private data. Indeed, information displayed on a wall is visible for all users standing in front of it. Von Zadow et al. proposed *SleeD* [25], that uses a mobile device attached to the user forearm to store data that can be displayed on the wall using bi-manual gestures.

In summary, a lot of work has been focusing on target pointing using either direct interaction or distant interaction from a desktop or using mobile devices, but a few on virtual navigation using interaction techniques in the context of wall-sized displays. That is why we focused our work on finding suitable techniques in a wall context as well as what interaction type was the best suited depending on the task performed.

CHAPTER 3

Virtual navigation

As we saw, interacting with a wall display can be performed in multiple ways and the type of technologies used in that process may have an impact on task performance. Thus we first had to decide what kind of systems we wanted to use. This choice was even harder considering the modalities we wanted to use. As an example, holding a mobile device makes difficult the use of bi-manual interaction on the wall; using mid-air gestures with a mobile device may lead to the *Midas problem* because interacting on a mobile device may be interpreted as mid-air gestures. We evaluated these technologies by imagining the techniques we could use, keeping in mind the goal of this internship: evaluate direct vs indirect interaction using the same techniques. We wanted the user to be able to switch easily from direct touch to remote interaction. We discuss these problems in this chapter.

3.1 Technologies

To compare direct vs indirect evaluation we had to combine at least two systems: a tactile system with a remote one. The wall display we had the opportunity to work on is the WILDER platform at Inria. It is a cluster of 10 computers controlling 7 to 8 21.6 inches LCD displays each. It is 5.90 meters wide and 1.96 meters high. It features narrow bezels[1] of less than 3mm. The resolution of such displays reaches a total resolution of 14400×4800 pixels (69 million pixels), thus offering very high display capacity. It is equipped with tactile sensors, audio rendering, and a 3D motion-tracking system.

TUIO

TUIO is a protocol used to track objects or fingers on a multitouch surface. It enables a quick and simple communication between a multitouch surface and a gesture recognizer. WILDER was built with this kind of system at the time, which was convenient for our study.

[1] distance between screens

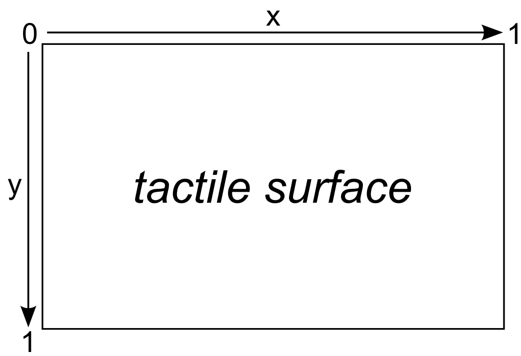


Figure 3.1: TUIO inputs interpretation

TUIO abstracts from the surface size and uses only *normalized coordinates*. So, positions x and y coordinate are included in [0; 1], representing a percentage of the width/height of the display. To achieve that on WILDER, the cluster of screens is surrounded by a frame composed of infra-red lights. When one's fingers cross the IR lights field, their positions are known and sent via wi-fi to a computer.

Even with a response time between 7-16 ms and a precision up to ±1.5mm, we had some detection problems with this system due mostly to the size of the display. As a matter of fact, using two fingers with a small gap between them or hovering the surface without touching it was considered as one input.

SMARTIES [6]

SMARTIES is a toolkit developped at LRI, which is focused on the interaction with a wall-size display at a distance. It is composed of an Android application coupled with a communication protocol to interact with a server-side application. When the application connects to the server, it updates its interface and behaviour. This system was exactly what we were looking for: it uses a tactile surface for all inputs, easing the adaptation of all techniques from direct to indirect interaction and is build specially to interact with a wall. Depending on the server-side application code, SMARTIES may feature the use of widgets, but we did not need this feature. Yet we decided to use this system to provide requested feedback [20], and be able to reproduce techniques exactly the same way.

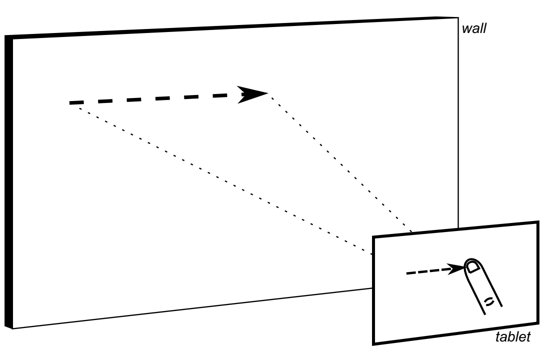


Figure 3.2: Ratio mobile device / wall

As we may see on Figure 3.2, display sizes have an impact on input. A movement on the tablet interpreted directly on the wall will be augmented by a factor. Knowing that, we needed to make a choice concerning the techniques. We had basically two options: either accelerate movements on the wall display or slow all movements coming from the mobile device.

As Figure 3.2 suggests, we faced an issue with that modality. A mobile device being a lot smaller than a wall, we had to proceed carefully and assure modalities will be comparable; in case of a favored modality, our results would have been biased.

VICON

VICON is a 3D motion capture system using infra-red cameras and marker-based technology. The current setting for WILDER is composed of 6 cameras tracking an area about 4×3 meters wide. A tracked object is made of infra-red markers precisely positioned in order to let the system know its geometry. It is therefore able to tell the orientation and position of the object relative to the bottom-center of the wall.

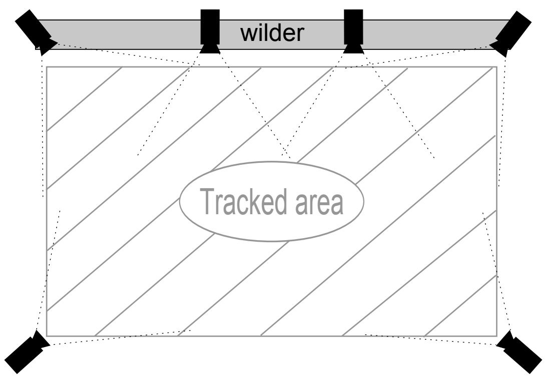


Figure 3.3: Vicon settings

This schema gives an idea about how the VICON system is tracking 3D motion. It covers almost the entire area in front of the wall with enough precision and a high enough refreshing rate to provide good interaction (respectively up to 0.5mm and 250 fps). As we have seen before, mid-air gesture is an option for distant interaction.

The VICON system is valuable for doing so, but adds precision constraints for the techniques we would like to test. Thus, we decided to only use it to track user position and give us a way to compare direct or indirect interaction in matters of physical navigation.

Now that we have seen the type of technologies used in the course of this internship, it is noteworthy that we did not choose them without knowing the techniques we wanted to use; neither the other way around. We had to iterate over the type of technologies and techniques used to find good matches.

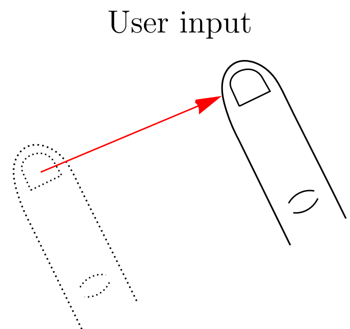
3.2 Interaction techniques

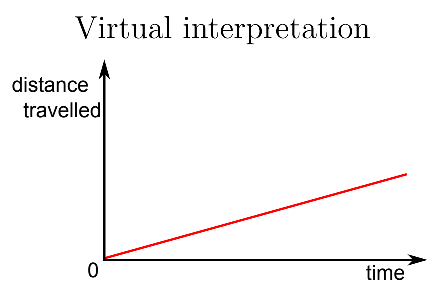
Wall-sized displays being still a new research area in Human-Computer Interaction, techniques designed specifically for them are not yet widespread. However, since we decided to use the IR lights frame combined with TUIO to perform direct interaction, we had the opportunity to use known techniques on tactile surfaces and adapt them to work on the wall. Of course, in order to do that, we had to take into account the size of the display as well as its vertical orientation. As we will see further, fatigue is an important factor to consider in the context of a wall display, so we used it as a selection argument.

Virtual navigation is made of two types of techniques : pan and zoom. Our task here was to find complementary techniques to avoid disturbances while using one or the other, and particularly be able to switch between them. It took us the first third of this internship to come up with adequate combinations. The first step was to list techniques, and the second to prototype them. We eliminated some techniques after prototyping them, as they were found to be exhausting or inefficient.

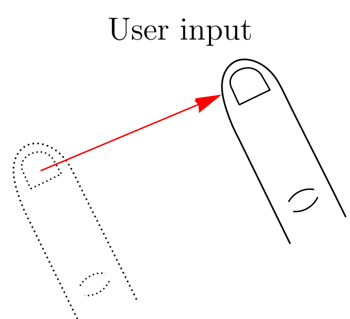
Pan techniques

Zero order pan is probably the most widespread and simple navigation technique available. It uses a zero order of control as defined by Zhai in his thesis [27], meaning that user input is not modified. The user has direct control of the virtual movement, which is convenient for precise tasks as placing a circle into another. This gesture is the basis of panning movements, making it difficult to avoid when talking about virtual navigation.
Tuning: for direct interaction we used users input without modify them, but for indirect interaction, as we saw, an absolute movement is not suited in that case so we defined a 1:1 ratio that mapped 1 pixel on the tablet to 1 pixel on the wall. Figures on the right show the use of this technique.

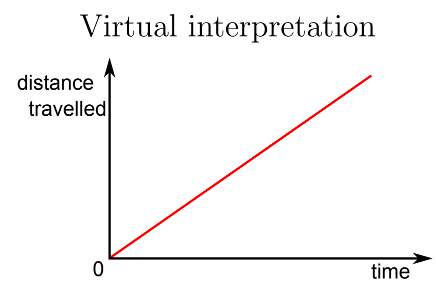




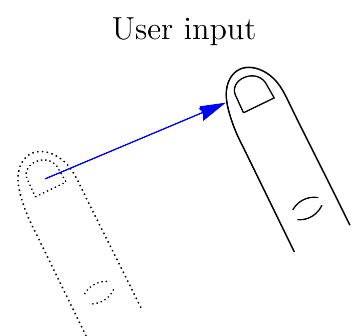
The **Gain** technique is exactly the same as *zero order pan*, but adds a defined factor to the movement. It is particularly useful if one wants to travel the whole width of the screen with only one gesture. This technique seemed interesting to investigate because of the control one has on it and the speed it provides.

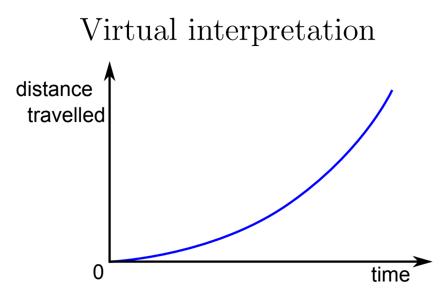


Tuning: for direct interaction we tuned it in order to cross the whole screen with only one gesture. To do that, we took an area picturing approximately what is within a user arm's reach (circle of diameter = 1m) and accelerated the movement by $\frac{width_{wall}}{width_{area}}$. For the distant interaction, we mapped the tablet directly to the whole display size. Thus, crossing the tablet resulted in crossing the wall display.

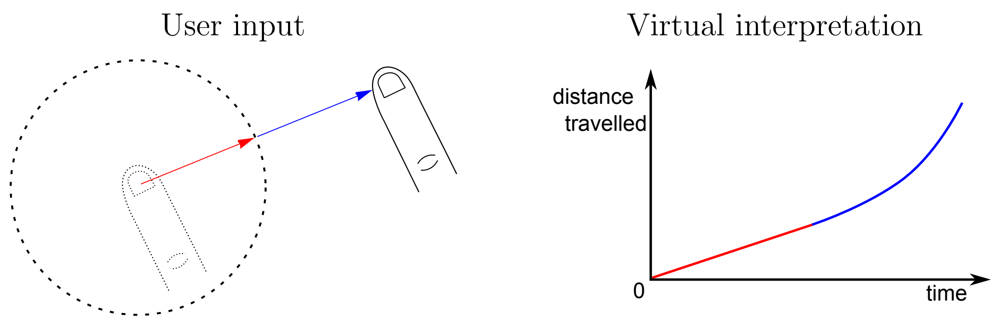


First order pan uses a first order of control, as defined again by Zhai in his thesis [27], meaning it acts on the speed of movement and not the amplitude. For this technique, the first user input defines the initial point of the vector speed. Moving the finger around will set the other extremity of the vector. Thus, the further one moves away from the initial point, the faster he will travel. This technique seemed suited for fast travel over great distances, but we wanted to know how users will use it (maybe even for small distances).
Tuning: for direct interaction we used user's input to set the speed vector; the speed depended on its norm. Using indirect interaction, we used a 1:1 ratio (like we did for *zero order pan*) on all inputs coming from the mobile device, to avoid favoring it over the direct interaction.

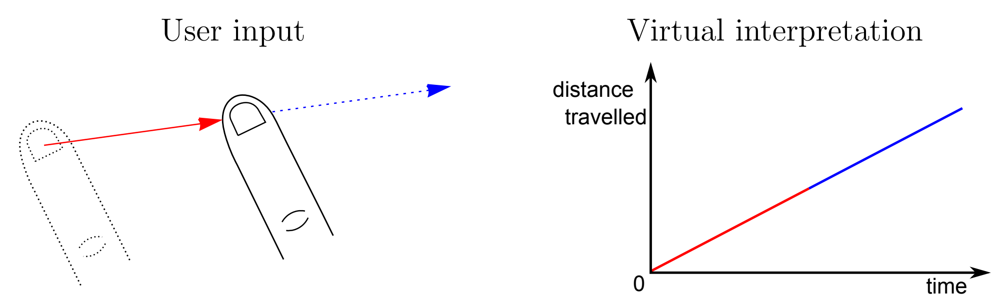




The **First order area** technique uses a *zero order pan* movement inside a given area and *first order pan* movement beyond. It is inspired by *RubberEdge*, presented by Casiez et al. [5], and tries to extend user reach on a wall. This technique was interesting because it was extending user reach, meaning that when the user had his arm almost straight, he could still interact by switching to first order of control and avoid clutching.
Tuning: we tuned this technique based on the same within arm-reach idea as the *gain* technique. The area into which the movement was absolute had a diameter equal to 75% of the *gain* area diameter (so 0.75m), letting the user switch easily between zero and first order of control. For the indirect interaction, the tablet used a 1:1 ratio like the *first order pan* technique, with an attenuation factor for first order of control to avoid losing control when switching from zero order to first order of control.



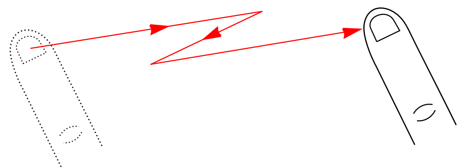
The **Flick** technique is a quick sweeping movement setting a translation speed. If the gesture is fast enough before releasing the finger from the interacting device, the virtual scene is translated according to the gesture's speed. We decided not to implement a dampening mechanism thinking it was more interesting to keep moving until the user stopped the movement. We found inspiration for this technique in *Flick-and-brake* proposed by Baglioni and al. [2], but decided not to use the braking feature: the TUIO system associated with WILDER could only recognize if the finger was pressed on the display, but could not measure finger pressure. We chose this technique because it seemed the best in terms of fatigue: being able to rest between the beginning and end of movement is well suited on a wall-sized display context.
Tuning: we used the last 6 input events to compute a speed vector. We then used a factor to normalize it and used a bound in case the flick was too strong. For indirect interaction we used a 1:1 ratio as usual because it was made of *zero order pan* extended with speed control feature.



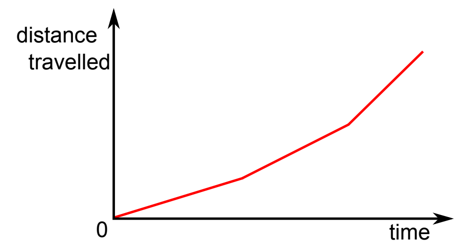
The **CycloPan** technique as described by Malacria et al. [15] is a clutch-free technique designed for wall-sized displays. It is based on elliptical gestures and lets the user change his gesture direction without changing the movement's direction, if he goes fast enough. Moreover, every time the user changes direction, it computes the last two stroke frequencies and changes the gain depending on them. We decided to investigate this technique for its clutch-free design and to see how users would

appreciate it.
Tuning: this technique had to use absolute pan movement below a given speed. We recorded the last 4 inputs of a user and tested the vector it defined : if its norm was greater than the threshold, we turned the *CycloPan* mode on. Since the speed of the movement is relative to the last stroke frequencies, we used the formula described in [15] to achieve that. Again, a 1:1 ratio was used for indirect interaction.

User input



Virtual interpretation

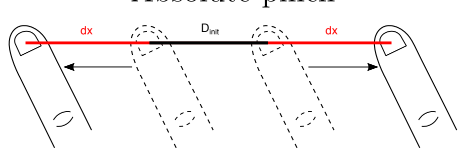


Zoom techniques

Panning techniques are useful for distances up to a threshold. Beyond that, zoom techniques are much more efficient. Usually, to reach a far target, the succession of actions follows the same pattern : zooming-out followed by sequences of zooming-in/panning actions for precise adjustment. We wanted to be able to switch easily between pan and zoom techniques: we describe below our choices.

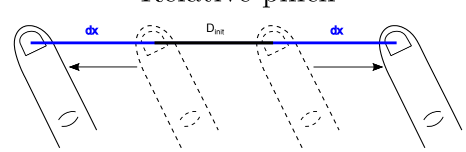
Pinch gestures are well known on mobile devices. They use two fingers to set the zoom factor and can enable fast zooming. In the context of a wall display, it may be preferable to use both hands to set the zoom factor, because of the wider possible movements. Interpretation of a pinch gesture may be done at least in two ways: the distance between the two fingers is either mapped to an absolute difference with the initial fingers' distance or a percentage of the initial fingers' distance. Both ideas are represented below.

Absolute pinch



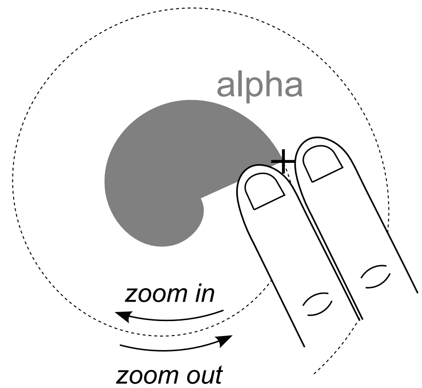
The new altitude is computed using $dx - D_{init}$

Relative pinch



The new altitude is computed using $\frac{dx}{D_{init}}$

CycloZoom is the sibling of *Cyclopan* as described by Malacria et al. [15]. It also uses elliptical gestures but map the zoom factor to the angular speed. Due to the Vernier effect [7], the further one moves away from the center of the ellipse, the slower the altitude changes. The advantage of such a technique combined with *Cyclopan* is to switch from pan to zoom gesture easily. The orientation of the zoom-in/zoom-out movements are based on a screw, respectively triggered by a clockwise/counter-clockwise movement.



Pan & Zoom

After listing all pan and zoom techniques, we had to combine them. We decided to use a pretty simple characterisation which was to split pan and zoom respectively to one and two fingers gestures. By doing this, we could achieve smooth transitions between pan and zoom as well as easy learning of the different techniques. We prototyped all techniques and ended-up with an important issue. As we will see, we had to build characterised tasks representing ecological tasks of virtual navigation, but did not find adequate trade-off between evaluate both zoom techniques and direct vs. indirect interaction. We finally decided to only evaluate pan techniques to have results that we could interpret and provide a basis for future work.

CHAPTER 4

User study

The need to evaluate new ideas in the computer science research field is essential. Focusing on Human-Computer Interaction, these evaluations take usually the shape of experiments involving humans. In some cases, the tasks studied may be really complex, spoiling data by introducing noise. Thus, they need to be operationalized, which consists in making them more abstract and simple to gather interpretable results. As an example, looking for geographical information on a map may be simplified to finding a target among many. We may then only display n targets on a wall and analyse the time a user will take to find a particular target. To be able to evaluate direct and indirect interaction on a wall-sized display, we had to go through a process of simplification to find the good task for our evaluation. Based on that we expected to find meaningful observations about our hypothesis.

4.1 Tasks

Depending on what a user wants to achieve and the options he has (like the number of hands he can interact with or the movement space available), his strategy may be completely different. To focus on navigation using a wall, we tried to build large scenes significantly exceeding the display to force users to navigate. The first scenes[1] we implemented were primarily a way to learn how to use ZVTM cluster [21], a Java library providing a lot of tools to draw vectorial graphics, and prototype techniques. Yet, we still used them to understand more clearly the main problem we were facing.

This scene took inspiration from the one in [22]; the point was to control all factors (distance between circles, sizes and the number of selection the user had to accomplish), and proposed another way to select a target we found more suited for virtual navigation : view pointing as described by Guiard et al. [10]. It made us also understand issues such as the “desert fog” problem [13].

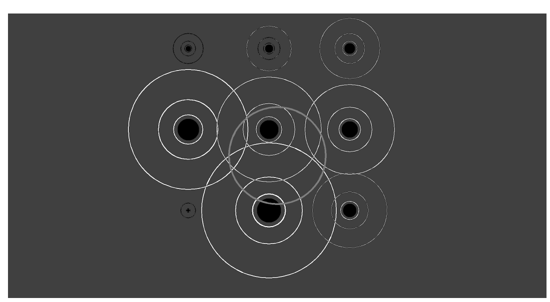


Figure 4.1: Scene prototyped

Implementing our first scenes made us think about two important issues. First,

[1] graphics environment displayed

the "desert fog" problem which occurs when rendering some shapes on a solid-color background. When losing visibility on the shapes, one gets disoriented and no longer knows where he is. This effect is amplified by the size of the display, it is therefore important to find a way to guide the user: we used concentric circles in this case (see Figure 4.1). Then, we needed a scene taking advantage of virtual navigation without being too complex otherwise we would have had too many factors to analyse.

We tried to focus on scenes involving more navigation and took inspiration from the scene described in [15]. The user had to select one target after the other, using pan and zoom techniques. But, evaluating direct vs. indirect interaction seemed still hard to perform in this case, because it could be basically performed using only zoom interactions.

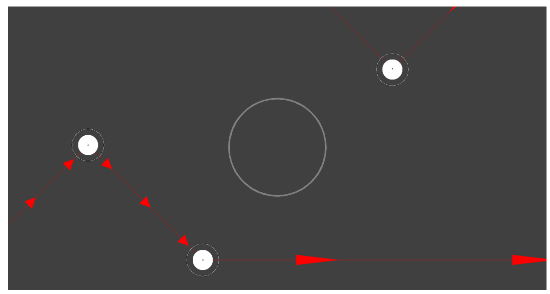


Figure 4.2: Navigation scene

With this scene, we faced another main problem which was to create a task using both pan and zoom without favoring one or the other. Moreover, we needed to find a task that could have advantages while using direct interaction over indirect interaction and vice-versa.

We ended-up prototyping a scene representing targets rendered as rectangles. Depending on the zoom factor, the targets were revealing information under a given threshold. The user had to look for a specific word and validate as soon as he found it. The word displayed in the middle of all targets was rendered with a very small font, forcing users to get close to the screen. This scene made us understand that evaluating direct/indirect interaction while enabling zooming was quite hard.

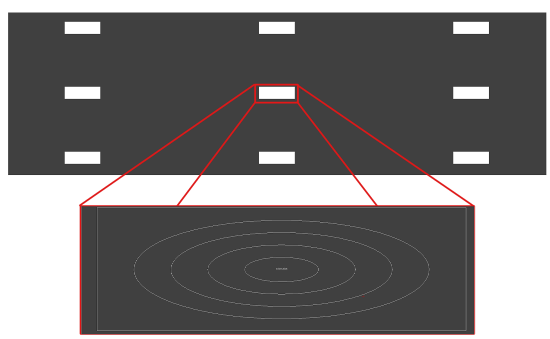


Figure 4.3: Details revealing

Pan only

Finally we focused on scenarios involving positioning an area, which was capturing the action a user would do when looking for detailed information about a location on a map for example. When doing so, the user looks for the direction in which he wants to pan, and then interacts to make the interesting area visible to him. This type of task can be simplified by putting a rectangle inside another one. We decided to split this task into two categories : a docking task capturing the need to focus on small details and a view pointing task capturing the need to have an overview of an area.

4.2 Experiment

Description

We designed an experiment and conducted in July. This experiment aimed at evaluating techniques using direct interaction on a wall display versus techniques using indirect interaction on a mobile device. We wanted to use the same techniques in order to find whether they were more adapted for one modality and particularly to have a way to compare types of interaction. As explained earlier, we wanted to investigate pan and zoom interaction but preferred to work on pan techniques which already had a lot of constraints. Moreover, we needed some results about direct vs. indirect interaction before proceeding further, and we did not find how to control properly zoom techniques in a way that would not bias our data.

From the techniques described above, we decided to use 5 of them : *Gain, CycloPan, First order, First order area, Flick*. We had to tune all techniques for all interaction types, the main problem being the one described in Figure 3.2. Each technique was implemented using one finger, exceptions made for the *Gain* and *First order* techniques for which users could use two fingers to trigger zero order of control.

We were also looking for information about user movements in front of the wall while interacting with it. To record their positions, we used a VICON system (described in Section 3.1). During the experiment, they were allowed to move freely. The only obligation they had was to stand up while interacting.

We had to create a parameterized context to be sure about our control of user tasks. After testing many alternatives, we ended up with the one shown in Figure 4.4. It was composed of two tasks:

The **docking task** is a selection of a small area of the entire display. It captures the need one could have when using virtual navigation on a wall-sized display to appreciate a detailed area of the information displayed. As an example, if we display a very high-resolution picture of Paris on the wall, one might need to move it to see details on top of it, therefore navigate to see it. We used two rectangles for this task, one representing the docking area and the other one a target to dock. During the experiment, the user had to put the target in a placeholder at the center of the display to successfully complete the trial (respectively b. and d. on Figure 4.4).

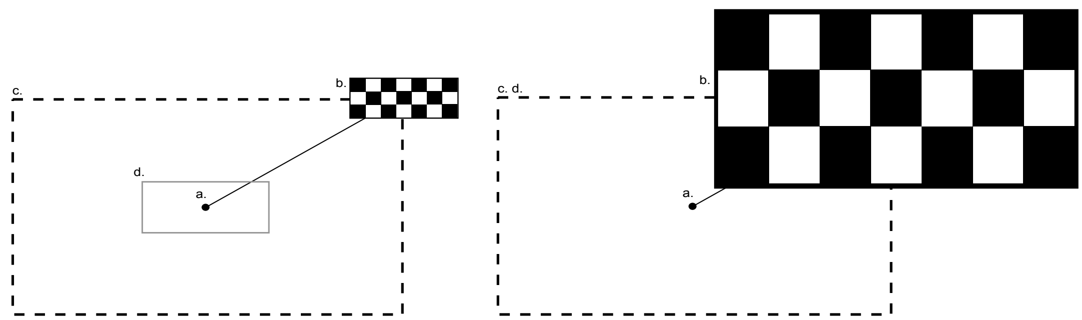


Figure 4.4: Docking (left) and View pointing (right) tasks : a) initial position of the user, b) target to dock/point, c) scene displayed on the wall, d) target's docking area

The **view pointing task** is based on the same idea but uses the whole screen as a docking area. The difference here is therefore not physical but mental; it is different in the goal we want to achieve. A view pointing task symbolizes the need to have an overview of a region. For example, one might want to see what a whole country looks like and what are the main cities that compose it. During the experiment, if the docking area was missing from the middle of the display, the user knew it was a view pointing task and had to see the whole target displayed before validating the trial.

There were several secondary factors : the **distance** of the target from the starting point of the trial (either half of the wall's size or the entire size), the **tolerance** of validation, defined as the gap between docking area bounds and target bounds (either 200px or 400px) and finally the **direction** toward which the target was displayed (top left, top right, bottom left or bottom right). Having these factors, we wanted to have a coarse idea of the overall time to complete the experiment. We computed the number of possible combinations and found $\#modalities \times \#techniques \times \#type \times \#distance \times \#tolerance \times \#direction = 320$ combinations which implies running an experiment about $320 \times 15 = 4800$ seconds (≈ 80 minutes) assuming users will take 15 seconds per trial. In the end, we expected an experiment lasting between 1 hour and 1 hour and a half.

Hypotheses

Before starting the experiment, we did some pilots to evaluate the hypotheses we had. From them, we were able to have a better idea of what results we might have and formulate the clearest hypotheses we thought were true. These were as follows :

H1- Using distant interaction is faster than direct interaction for view pointing tasks

H2- Docking tasks are faster than view pointing tasks (mostly due to the same tolerances)

H3- The distance to the target affects task time performance

H4- Using indirect interaction with a tablet is less tiring than interacting directly on the wall

Results

We ran the experiment on 10 users (6 men and 4 women) aged from 23 to 33 years old, all using tactile surfaces daily. They performed the experiment on WILDER (details in Section 3.1 above). The experiment was split in two parts, using one type of interaction after the other. We used two latin squares to order all techniques respectively for direct and indirect interaction. The first 5 users did the experiment starting with direct interaction and finishing with indirect interaction, and the last 5 did it the other way around. At the end of the experiment, all users answered a questionnaire containing Likert-scale evaluations about the techniques. First they had to rank them from 1 to 10 with 1 being the most preferred and 10 the least, in the context of docking tasks and view pointing tasks. Then evaluate them from 1 to 5 with 1 being the less tiring and 5 the most (see Annexe A).

Statistics

Our main interest was toward the comparison of direct and indirect interaction. In order to do that, we ran an ANOVA test[2] (results are shown in Table 4.1) on all factors in the experiment and found an interaction effect ($p = 0.0018$ with a great effect size $\eta^2_G = 0.26$) on the overall time to complete a trial. We then ran a post-hoc test showing that indirect interaction is significantly (p = 0.002) faster than direct interaction, validating **H1** (see Figure 4.5). Looking at the same ANOVA test results, we found an effect of the task type on time ($p < 0.0001$ and $\eta^2_G = 0.19$). After running a post-hoc test, we were able to say that a docking task is significantly faster (p = 0.0001) than a view pointing task, no matter the interaction type used, validating **H2**. Yet, the view pointing task had more different results when using direct or indirect interaction than docking task (p = 0.0011).

As we predicted, the distance to the target from the starting point of each trial had an effect on task completion time. We found an effect ($p < 0.001$ and $\eta^2_G = 0.133$) that was confirmed by a post-hoc test ($p < 0.001$) validating **H3**. Even an interaction

[2] Overall, we had an error rate of 6%. Thus, for all tests we made, we decided not to take into account trials with errors

between the interaction type used and the distance was detected by the ANOVA test ($p = 0.004$ with $\eta_G^2 = 0.038$).

Looking at the distance travelled by a user while performing a trial, we also found some effects of the interaction type used or the type of task performed. As we may see in Table 4.1, the interaction type had an effect on the distance travelled ($p = 0.0016$ and a great effect size $\eta_G^2 = 0.43$) as well as the type of task performed ($p = 0.0080$ and $\eta_G^2 = 0.03$). There is no need to run post-hoc tests here since we only have two values for these factors (see Figure 4.7).

If we now turn to the comparison of all techniques, we may have two main points of view. First, we want to compare the efficiency of all techniques used, and after that evaluate participants' subjective preferences. From the previous ANOVA test results (Table 4.1), we had also an effect of the technique used on the time to complete each trial ($p < 0.0001$ and $\eta_G^2 = 0.19$). The post-hoc test revealed that the *Flick* and *Gain* techniques were the fastest compared to all others (see Figure 4.6).

Finally we wanted to have users feeling about the techniques they just used. The Figure 4.8 summarizes user answers. We ran another ANOVA test (see results in Table 4.2) on the data and found an effect of the techniques ($p < 0.0001$ and $\eta_G^2 = 0.48$) and interaction type ($p = 0.0221$ and $\eta_G^2 = 0.02$) on users' preferences. Running a post-hoc test showed us that *CycloPan* was rated worst (maximum $p = 0.001$) and indirect interaction was rated best ($p = 0.022$). The device as well as the techniques had an effect on fatigue. As we can see in Table 4.2, we detected effects of the technique ($p < 0.0001$ and $\eta_G^2 = 0.32$) and interaction ($p = 0.0402$ and $\eta_G^2 = 0.09$) on the fatigue felt by the users. Running a final post-hoc test showed us that *CycloPan* was felt as the most tiring technique and direct interaction the most tiring interaction type, validating **H4**.

	Time to Complete the Trial				Distance travelled			
Factor	n,d	$F_{n,d}$	p	η^2_G	n,d	$F_{n,d}$	p	η^2_G
Technique	4,36	10.8	< 0.0001	0.19	4,36	3.15	0.0254	0.03
Interaction	1,9	18.9	0.0018	0.26	1,9	19.7	0.0016	0.43
Task	1,9	44.9	< 0.0001	0.19	1,9	11.4	0.0080	0.03
Technique ×Interaction	4,36	0.60	0.6629	< 0.01	4,36	3.33	0.0202	0.03
Technique ×Task	4,36	1.79	0.1516	< 0.01	4,36	2.04	0.1089	< 0.01
Interaction ×Task	1,9	24.67	0.0007	0.04	1,9	4.62	0.0598	0.01
Technique ×Interaction ×Task	4,36	1.12	0.3621	< 0.01	4,36	4.31	0.0050	< 0.01

Table 4.1: ANOVA test looking for effects between Techniques, Interaction types or Task types and the Time to Complete a Trial (left) or the Distance travelled (right)

	Users preferences				Fatigue felt			
Factor	n,d	$F_{n,d}$	p	η^2_G	n,d	$F_{n,d}$	p	η^2_G
Technique	4,36	26.8	< 0.0001	0.48	4,36	10.4	< 0.0001	0.32
Interaction	1,9	7.61	0.0221	0.02	1,9	5.73	0.0402	0.09
Task	1,9	2.26	0.1663	< 0.01		—	—	
Technique ×Interaction	4,36	1.91	0.1282	0.01	4,36	0.93	0.4546	< 0.01
Technique ×Task	4,36	0.31	0.8635	< 0.01		—	—	
Interaction ×Task	1,9	1.76	0.2165	< 0.01		—	—	
Technique ×Interaction ×Task	4,36	0.36	0.8313	< 0.01		—	—	

Table 4.2: ANOVA test comparing Technique, Interaction and Task types to user preferences (left) and fatigue felt (right)

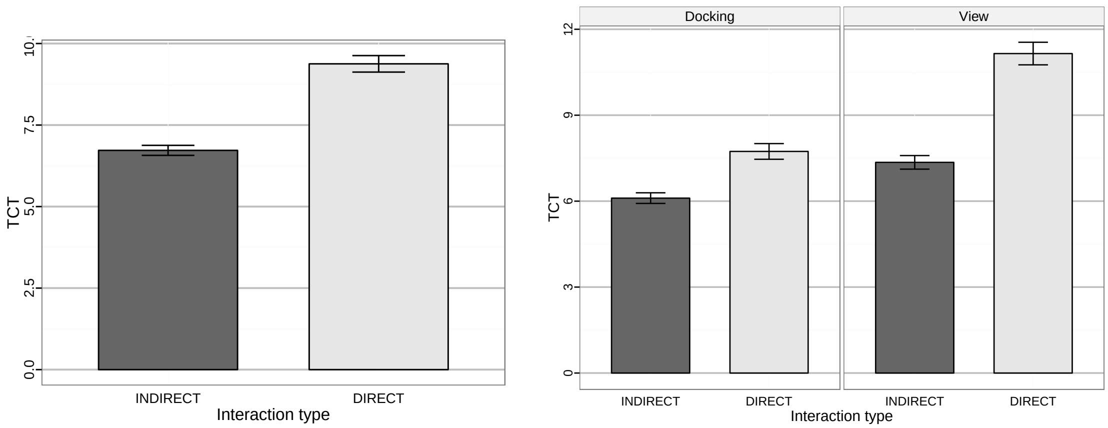


Figure 4.5: Mean time for the interaction used (left); taking task type into account (right)

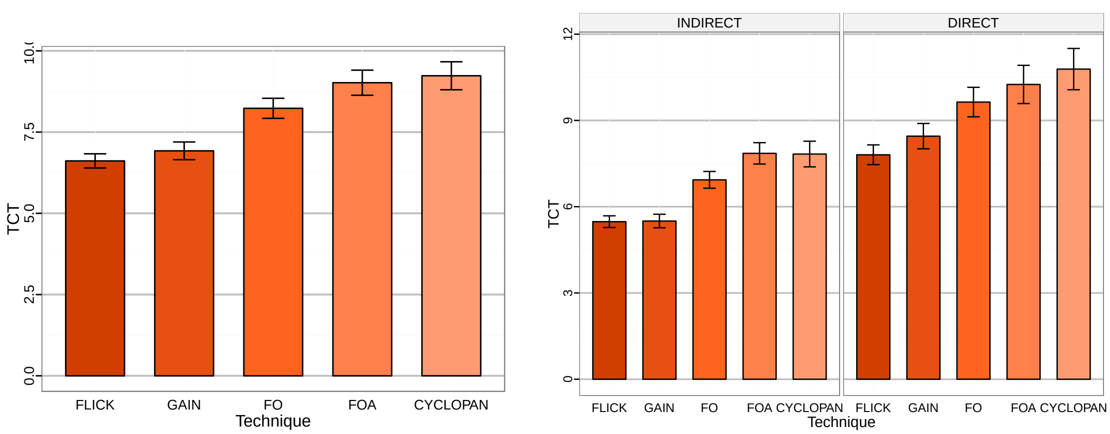


Figure 4.6: Mean time for the technique used (left); taking interaction type into account (right)

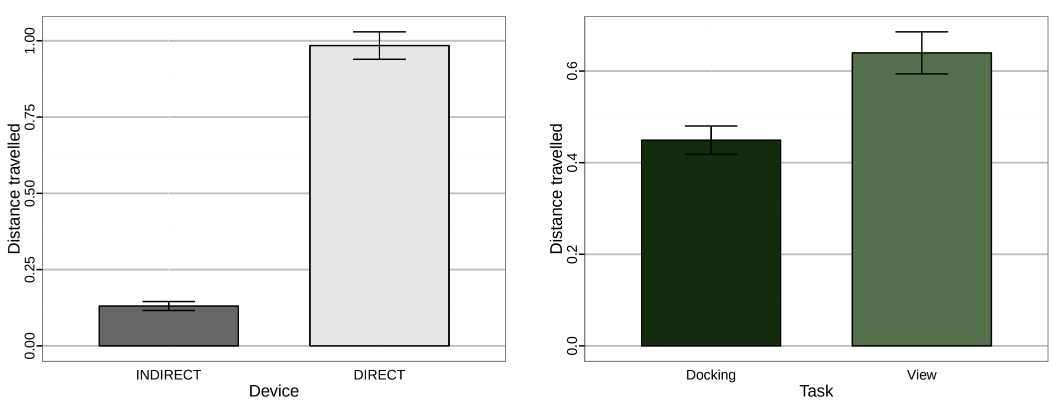


Figure 4.7: Distance travelled relative to the interaction type (left) or the task type (right)

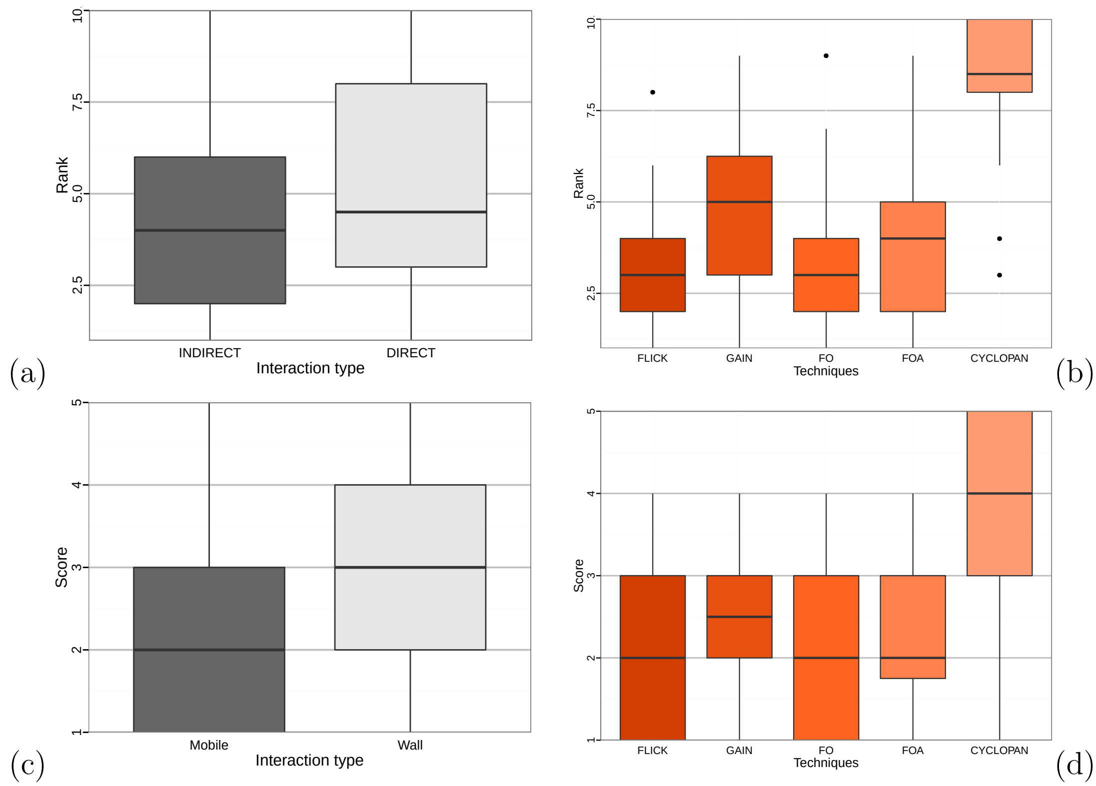


Figure 4.8: (a) interaction types preferences, (b) techniques preferences, (c) fatigue according to the interaction types, (d) fatigue according to the techniques

Interpretations

Indirect interaction is faster than direct interaction was an expected result because when interacting directly on the wall, users do not have an overview of the display, making view pointing tasks more difficult. However, we did not expect to observe faster performance using indirect interaction for the docking task. Our tentative explanation for these results is that when interacting directly on the display, it is harder to predict when the target will be within arm's reach while the scene is moving. These results led us to design two follow-up experiments briefly described in Chapter 5.

Distance to the target has an effect on task completion time was expected. Indeed, the further the target is, the more time one will take to reach it. An interesting result is that using the tablet, users did take less time to reach the targets. We may assume this is mostly due to the overview one has from afar, therefore losing less time to recalibrate a missed trajectory movement toward the target.

The Flick and Gain techniques had the best performance overall. The results about techniques are not surprising, both techniques allow a fast movement with a great control. An important detail here was that we did not find an effect of the techniques over the interaction type, meaning that the techniques did not perform better with direct or indirect interaction. We may assume that they were well tuned to be as similar as possible for the experiment, and using one technique or the other on a wall will not be significantly different from using it on a tablet.

Users do not move while using distant interaction is an interesting result for our experiment. At a distance, one does not need to move but only rotate his head, whereas when interacting directly on the display it may be necessary to check if the target is placed correctly into the docking area, especially for the view pointing task. Thus, we can say that while interacting on the wall, users tend to move more, and even more if they have to perform a view pointing task.

Direct interaction is more tiring than indirect interaction was an expected result. Of course, when interacting directly on the wall, one has to hold his arm up, which is more tiring than holding a tablet to interact with it. However, we expected the *Flick* technique to be the preferred technique because it answers the fatigue problem on the wall, but we did not find that.

Users remarks

Some users did make interesting remarks leading the same way as our future work. Two users said that using touch interaction directly on the wall made them feel like they were "carrying" the target. This is the kind of effect that makes an interaction process soft, enjoyable, and sometimes even more efficient. As described below, we want to investigate that feeling with a follow-up experiment. Another remark we had more than once was about the questionnaire. One user found it hard to rank all techniques at the end of the experiment, because she had to remember her first trials. Another one asked if there will be questions at the end of the experiment (10 minutes after starting it), and if he could hear them to think about his answers precisely. This is a delicate issue to solve; on one hand if they now what will be the questionnaire at the end, they will only look for answers that way, but on the other hand, it is hard to remember exactly how you felt about a technique at a precise moment.

Chapter 5

Conclusion and perspectives

The results we found with that experiment revealed that indirect interaction seems to be the most suited interaction for panning tasks on a wall-sized display. We expected to have better results using direct interaction for docking tasks and were surprised by these. We decided to design a follow-up experiment composed only of docking tasks using the *Flick* technique, but split indirect interaction in two: within arm's reach of, or 2 meters away from, the display. The idea is to understand if the only advantage of using a mobile device is to have an overview of the display. Moreover we designed another one to investigate the advantage of direct over indirect interaction for small distances. To do that, the follow-up consists of docking tasks using the *Flick* technique but only within arm's reach distances. The interaction types were direct and indirect. The idea is to show that direct touch is more efficient for precise absolute movement because the user has the feeling of grabbing the target. Unfortunately, we did not have the time to run any of them before submitting this report, but we should before the end of the internship.

The results also showed us that the *Flick* and *Gain* techniques are performing best in the context of panning tasks on a wall-sized display using both direct or indirect interaction. Knowing that, we could use our results to integrate the *Flick* and *Gain* techniques in systems such as *SleeD* [25] to improve map navigation. Our future work will probably focus on zoom techniques and reproduce the same type of experiment to evaluate their performance using direct or indirect interaction. In the end, our goal is to propose efficient techniques to navigate virtually in a wall-sized display context combined with an efficient interaction system.

To conclude, this internship was a great experience in terms of learning and research matter. I had the opportunity to learn how to design and conduct an experiment from scratch and gather meaningful observations about a particular study as well as analyse statistically the results found using the R language. We will hopefully end this internship by a submission at CHI '15, the flagship conference in HCI. Finally, working in the ILDA team with Olivier and Emmanuel has been a real pleasure, and hopefully will stay the same in the future.

Bibliography

[1] Christopher Andrews, Alex Endert, and Chris North. Space to think: Large high-resolution displays for sensemaking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 55–64, New York, NY, USA, 2010. ACM.

[2] Mathias Baglioni, Sylvain Malacria, Eric Lecolinet, and Yves Guiard. Flick-and-brake: Finger control over inertial/sustained scroll motion. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, pages 2281–2286, New York, NY, USA, 2011. ACM.

[3] Robert Ball, Chris North, and Doug A. Bowman. Move to improve: Promoting physical navigation to increase user performance with large displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 191–200, New York, NY, USA, 2007. ACM.

[4] Amartya Banerjee, Jesse Burstyn, Audrey Girouard, and Roel Vertegaal. Pointable: An in-air pointing technique to manipulate out-of-reach targets on tabletops. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '11, pages 11–20, New York, NY, USA, 2011. ACM.

[5] Géry Casiez, Daniel Vogel, Qing Pan, and Christophe Chaillou. Rubberedge: Reducing clutching by combining position and rate control with elastic feedback. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, UIST '07, pages 129–138, New York, NY, USA, 2007. ACM.

[6] Olivier Chapuis, Anastasia Bezerianos, and Stelios Frantzeskakis. Smarties: An input system for wall display development. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, pages 2763–2772, New York, NY, USA, 2014. ACM.

[7] Kenneth B. Evans, Peter P. Tanner, and Marceli Wein. Tablet-based valuators that provide one, two, or three degrees of freedom. In *Proceedings of the 8th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '81, pages 91–97, New York, NY, USA, 1981. ACM.

[8] Paul M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. In *Journal of Experimental Psychology*, pages 381–391.

[9] Clifton Forlines, Daniel Vogel, and Ravin Balakrishnan. Hybridpointing: Fluid switching between absolute and relative pointing with a direct input device. In

Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology, UIST '06, pages 211–220, New York, NY, USA, 2006. ACM.

[10] Yves Guiard and Michel Beaudouin-Lafon. Target acquisition in multiscale electronic worlds. *Int. J. Hum.-Comput. Stud.*, 61(6):875–905, December 2004.

[11] Robert J.K. Jacob. Eye movement-based human-computer interaction techniques: Toward non-command interfaces.

[12] Mikkel R. Jakobsen, Yvonne Jansen, Sebastian Boring, and Kasper Hornbæk. Should i stay or should i go? selecting between touch and mid-air gestures for large-display interaction (to appear). INTERACT '15, 2015.

[13] Susanne Jul and George W. Furnas. Critical zones in desert fog: Aids to multiscale navigation. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, UIST '98, pages 97–106, New York, NY, USA, 1998. ACM.

[14] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, Eric Lecolinet, and Wendy E. Mackay. Effects of display size and navigation type on a classification task. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, pages 4147–4156, New York, NY, USA, 2014. ACM.

[15] Sylvain Malacria, Eric Lecolinet, and Yves Guiard. Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces: The cyclostar approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 2615–2624, New York, NY, USA, 2010. ACM.

[16] Shahzad Malik, Abhishek Ranjan, and Ravin Balakrishnan. Interacting with large displays from a distance with vision-tracked multi-finger gestural input. In *ACM SIGGRAPH 2006 Sketches*, SIGGRAPH '06, New York, NY, USA, 2006. ACM.

[17] Jörg Müller, Gilles Bailly, Thor Bossuyt, and Niklas Hillgren. Mirrortouch: Combining touch and mid-air gestures for public displays. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services*, MobileHCI '14, pages 319–328, New York, NY, USA, 2014. ACM.

[18] Brad A. Myers, Rishi Bhatnagar, Jeffrey Nichols, Choon Hong Peck, Dave Kong, Robert Miller, and A. Chris Long. Interacting at a distance: Measuring the performance of laser pointers and other devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, pages 33–40, New York, NY, USA, 2002. ACM.

[19] Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P. Irani, and Michel Beaudouin-Lafon. High-precision pointing on large wall displays using small handheld devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 831–840, New York, NY, USA, 2013. ACM.

[20] Mathieu Nancel, Julie Wagner, Emmanuel Pietriga, Olivier Chapuis, and Wendy Mackay. Mid-air pan-and-zoom on wall-sized displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 177–186, New York, NY, USA, 2011. ACM.

[21] Emmanuel Pietriga. A toolkit for addressing hci issues in visual language environments. *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 00:145–152, 2005.

[22] Emmanuel Pietriga, Caroline Appert, and Michel Beaudouin-Lafon. Pointing and beyond: An operationalization and preliminary evaluation of multi-scale searching. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 1215–1224, New York, NY, USA, 2007. ACM.

[23] Constantin Schmidt, Jörg Müller, and Gilles Bailly. Screenfinity: Extending the perception area of content on very large public displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 1719–1728, New York, NY, USA, 2013. ACM.

[24] Garth Shoemaker, Anthony Tang, and Kellogg S. Booth. Shadow reaching: A new perspective on interaction for large displays. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, UIST '07, pages 53–56, New York, NY, USA, 2007. ACM.

[25] Ulrich von Zadow, Wolfgang Büschel, Ricardo Langner, and Raimund Dachselt. Sleed: Using a sleeve display to interact with touch-sensitive display walls. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces*, ITS '14, pages 129–138, New York, NY, USA, 2014. ACM.

[26] Colin Ware and Marlon Lewis. The dragmag image magnifier. In *Conference Companion on Human Factors in Computing Systems*, CHI '95, pages 407–408, New York, NY, USA, 1995. ACM.

[27] Shumin Zhai. *Human Performance in Six Degree of Freedom Input Control*. PhD thesis, University of Toronto, 1995.

APPENDIX A

Questionnaire

Overall appreciation

For the docking task did you overall prefer to use : *

- ◯ Touch on wall
- ◯ Touch on mobile device
- ◯ Both

For the view pointing task did you overall prefer to use : *

- ◯ Touch on wall
- ◯ Touch on mobile device
- ◯ Both

Overall the tasks were more tiring using : *

- ◯ Touch on wall
- ◯ Touch on mobile device
- ◯ Both

Fatigue appreciation

For all techniques rank the fatigue you felt from 1 to 5

Gain on Wall *

	1	2	3	4	5	
Not tiring at all	◯	◯	◯	◯	◯	Very tiring

Gain on Mobile *

	1	2	3	4	5	
Not tiring at all	◯	◯	◯	◯	◯	Very tiring

Cyclopan on Wall *

	1	2	3	4	5	
Not tiring at all	◯	◯	◯	◯	◯	Very tiring

Cyclopan on Mobile *

	1	2	3	4	5	
Not tiring at all	◯	◯	◯	◯	◯	Very tiring

Interaction appreciation

For the docking task rank the techniques from 1 to 10

Gain on Wall *

	1	2	3	4	5	6	7	8	9	10	
Preferred	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Least preferred

Gain on Mobile *

	1	2	3	4	5	6	7	8	9	10	
Preferred	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Least preferred

Cyclopan on Wall *

	1	2	3	4	5	6	7	8	9	10	
Preferred	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Least preferred

Cyclopan on Mobile *

	1	2	3	4	5	6	7	8	9	10	
Preferred	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Least preferred

For the view pointing task rank the techniques from 1 to 10

Gain on Wall *

	1	2	3	4	5	6	7	8	9	10	
Preferred	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Least preferred

Gain on Mobile *

	1	2	3	4	5	6	7	8	9	10	
Preferred	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Least preferred

Cyclopan on Wall *

	1	2	3	4	5	6	7	8	9	10	
Preferred	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Least preferred

Cyclopan on Mobile *

	1	2	3	4	5	6	7	8	9	10	
Preferred	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Least preferred