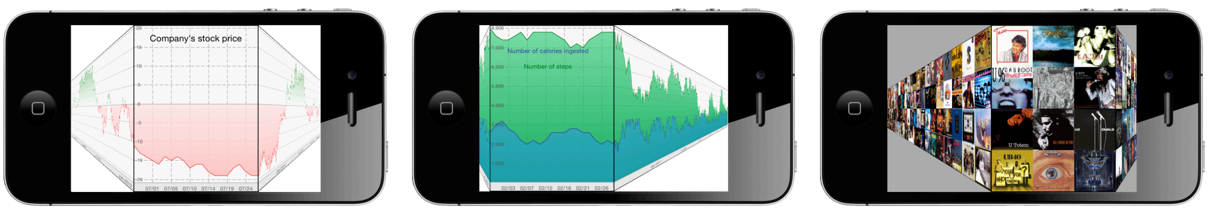
Multimodal Interaction with a Bifocal View on Mobile Devices

Sebastien Pelurson[1,2] and Laurence Nigay[1]

[1]Univ. Grenoble Alpes, LIG, CNRS
F-38000 Grenoble, France
{Sébastien.Pelurson, Laurence.Nigay}@imag.fr

[2]Motwin Company
F-38240 Meylan, France
Sebastien.Pelurson@motwin.com



(a) (b) (c)

Figure 1. Implemented bifocal display on iPhone 4. Three examples of applications: (a) Financial market: Stock prices. (b) Health&Well-being: Comparison of the numbers of steps and calories ingested. (c) Music library: Album covers.

ABSTRACT
On a mobile device, the intuitive Focus+Context layout of a detailed view (focus) and perspective/distorted panels on either side (context) is particularly suitable for maximizing the utilization of the limited available display area. Interacting with such a bifocal view requires both fast access to data in the context view and high precision interaction with data in the detailed focus view. We introduce combined modalities that solve this problem by combining the well-known flick-drag gesture-based precise modality with modalities for fast access to data in the context view. The modalities for fast access to data in the context view include direct touch in the context view as well as navigation based on drag gestures, on tilting the device, on side-pressure inputs or by spatially moving the device (dynamic peephole). Results of a comparison experiment of the combined modalities show that the performance can be analyzed according to a 3-phase model of the task: a focus-targeting phase, a transition phase (modality switch) and a cursor-pointing phase. Moreover modalities of the focus-targeting phase based on a discrete mode of navigation control (direct access, pressure sensors as discrete navigation controller) require a long transition phase: this is mainly due to disorientation induced by the loss of control in movements. This effect is significantly more pronounced than the articulatory time for changing the position of the fingers between the two modalities ("homing" time).

Categories and Subject Descriptors
H.5.2 [**Information interfaces and presentation (e.g., HCI)**]: User Interfaces.

General Terms
Design, Human Factors.

Keywords
Mobile devices; Bifocal view; Multimodal interaction.

ICMI 2015, November 9–13, 2015, Seattle, WA, USA.
© 2015 ACM. ISBN 978-1-4503-3912-4/15/11…$15.00.
DOI: http://dx.doi.org/10.1145/2818346.2820731

1. INTRODUCTION
As mobile technology increasingly becomes a very common mode of access to information and services, one challenge is be able to manage large information spaces on mobile devices. Interactive visualization on mobile devices addresses the problem of large information spaces and small displays. Although mobile displays increase in size and resolution (e.g., iPhone 6 Plus, 5.5", 1920 x 1080 pixels), information spaces are increasing more rapidly.

Several studies have focused on the interactive visualization of spatial and tabular information structures on mobile devices and in particular mobile maps [19, 23, 29]. We note that nowadays 1D and temporal information structures are increasingly managed on mobile devices, including financial data and more recently health and wellness data (number of steps, calories ingested, etc.). Figure 1 includes three examples of applications. Addressing this challenge we focus on the interactive visualization of 1D/temporal information structure on mobile devices.

Facing a large information space, zooming interfaces display the data at one level of detail at a given time and the user must navigate between the levels of data by zooming because of the display space limitation. As an alternative to pan&zoom navigation, the Overview+Detail and Focus+Context interfaces simultaneously display the data at multiple levels of detail. Overview+Detail interfaces simultaneously display two spatially separate views, each view containing data at a given level of detail. Overview+Detail interfaces on mobile devices are explored by Burigat&Chittaro [7]. Avoiding the mental effort to relate the two separate views, Focus+Context interfaces integrate the detailed view (focus) within the context view that displays the rest of the data with less detail. This paper focuses on bifocal display on mobile devices, one type of Focus+Context interfaces with two levels of detail as illustrated in Figure 1.

The paper introduces input modalities for interacting with a bifocal visualization on mobile devices. The addressed problem is to provide both a fast access to data in the context view and high precision interaction with data in the detailed focus view as described by Appert et al. [2]. Our design approach relies on the combination of modalities. Designed combined modalities are made of one modality for fast access to data in the context view and one modality for precise interaction in the detailed focus view. The results of an experimental evaluation of several designed combined modalities show that the performance can be analyzed according to a 3-phase model of the task: a focus-targeting phase (interaction in the context view), a transition phase (modality switch) and a cursor-pointing phase (interaction in the focus view). The performance of the observed navigation strategy is mainly based on the time of the focus-targeting phase and the time of the transition phase. Furthermore modalities based on a discrete mode of navigation control (direct access, pressure sensors as discrete navigation controller) require a long transition phase for switching to the precise modality. Due to the disorientation induced by the loss of full continuous control over the movements, this effect is significantly more pronounced than the articulatory time for changing the position of the fingers between the two modalities.

2. IMPLEMENTED BIFOCAL VIEW

Several Focus+Context techniques have been designed and empirically evaluated on a desktop PC such as the one described by Kincaid [18]. But such experimental results no longer hold on mobile devices because here specific constraints are present including the small size of the screen and instability due to mobility as described by Chittaro [9]. For instance Jakobsen&Hornbæk [16] compared of a Focus+Context technique on three different sizes of display and revealed the inferiority of a small display both in terms of task completion time and subjective assessments of effort. Moreover Gutwin [13], found that navigating on a small screen is slower than on a normal screen.

On mobile devices, Focus+Context techniques have been designed [8, 11, 29], exploring different types of information structures and application domains, including web browsers, scatterplots and maps. For instance the fisheye calendar DateLens presented by Bederson & al. [5] was found superior to an existing calendar. Moreover, although Jakobsen&Hornbæk [16] showed poor performance for a Focus+Context technique, the experimental comparison of different visualization techniques (Pan&Zoom, Overview+Detail, Focus+Context) in [8, 13] showed the viability of a Focus+Context technique for a 2D information structure, highlighting the value of the navigational context on mobile devices. We therefore chose one type of Focus+Context techniques that we study for 1D information structure: a bifocal display in a one-dimensional form (Figure 1).

The Focus+Context technique described by Huot&Lecolinet [14] is dedicated to 1D linear data more suitable for periodic data or for data without much detail. Indeed, most of the screen space is used by the context, so the focus area is small. As opposed to Huot&Lecolinet [14] and based on the requirements of our studied application domains (Figure 1), we designed and implemented a bifocal display with a large detailed view and two distorted sideviews (Figure 1). The display then manages two distinct levels of detail.. While preserving spatial continuity, the drawback is the perceivable discontinuity at the boundaries between the detailed view and the distorted views as described by Leung&Apperley [20]. Providing a smooth transition between the detailed view and the distorted sideviews, the Perspective Wall presented by Mackinlay et al. [22] is a generalization of the bifocal display that implements a non-constant demagnification function. Our goal was to evaluate the input modalities so we implemented a constant demagnification. Nevertheless from the Perspective Wall we reused the intuitive 3D perspective that enables us to display more information in the context area as described by Mackinlay et al. [22]. In the implemented bifocal display, the context is therefore bent perspectively as done by Mackinlay et al. [22] (Figure 1). The user defines the ratio of detail and context by resizing the focus, as done by Mackinlay et al. [22]. We implemented this functionality on the mobile phone with a resizing pinch gesture performed in the focus area. As with the Document Lens of Robertson&Mackinlay [25] and in order to maximize the utilization of the available display area (avoiding the waste of screen real estate, as with the Perspective Wall, when the detailed view, fixed at the center, is at the edges of the information space), the detailed view is not fixed at the center and is moved by users in order to explore the dataset. In the following sections we focus on the input modalities for moving the detailed focus view (i.e., navigation in the context view) and for precise interaction in the detailed focus view.

3. RELATED WORK: NAVIGATION MODALITIES

We restrict our review of related work to navigation modalities on mobile devices. Indeed the main elementary task for a Focus+Context visualization technique is to move the focus area in order to obtain the details of data belonging to the context area.

The first modality is the flick-drag scrolling. It is the traditional input modality commonly used today on mobile devices. For a large information space, the flick-drag scrolling modality can be fastidious and long as explained by Spindler et al. [28]. Moreover cumulative gain across flick gestures (as described by Quinn et al. [24] for iOS) can lead to a very fast scrolling speed that may imply a complete loss of control of the movements.

On the one hand to reduce clutching to cover long distances, the Flick-and-Brake technique of Baglioni et al. [4] is based on flick scrolling but allows the user to control speed deceleration by pressing a finger on screen. Without touching the screen, the motion continues.

On the other hand, several other modalities have been explored and are based on different sensors embedded on the mobile device. The goal of these modalities is to improve the efficiency of navigation but also to avoid finger gestures that cause occlusion of the displayed dataset while navigating and require users to perform switches between navigation mode and edit mode as explained by Spelmezan et al. [27].

- The technique described by Kratz et al. [19] extends the SDAZ technique developed by Igarashi&Hinckley [15] for mobile map navigation using tilt as an input modality. The authors compared it to a standard multitouch modality. The results show that the tilt modality performed at least as well as the multitouch modality. For one-dimensional navigation tasks, the tilt scrolling is compared to the flick scrolling by Fitchett&Cockburn [10]. The results show that tilt scrolling outperforms flick scrolling in a stationary situation. Nevertheless the focus was on reading and analysis tasks. Such tasks involve slow scrolling in short distance navigation tasks. Complementary to this study, we focus on the interaction task only as done by Appert&Fekete [3], without considering the perception task. In addition we consider a large information structure involving long distance tilt navigation tasks.
- SidePress by Spelmezan et al. [27] provides users with navigation capabilities based on pressure sensors located on the side of the mobile device. A comparison with the drag-flick modality shows that the pressure-based modality can be more efficient than finger gesture-based modality. In our study, we consider modalities based on two continuous

pressure sensors located on the top side of the mobile phone used in landscape mode.

- Two studies explore another modality based on the spatial manipulation of the mobile device. Pahud et al. [23] and Spindler et al. [28] compare the spatial manipulation with the standard Pinch-Drag-Flick technique (touch modality). In [23] results show that the spatial manipulation performs as well as the standard modality, whereas Spindler et al. [28] showed that spatial manipulation outperforms the standard modality. All these studies are based on a spatial information structure. In our study on one-dimensional navigation tasks we consider a spatial manipulation-based input modality.

4. DESIGN OF COMBINED MODALITIES

The design of modalities for interacting with a bifocal view on mobile devices is driven by the need for both a fast access to data in the context view and high precision interaction with data in the detailed focus view. To do so and as explained by Appert et al. [2], there are two ways to interact with the bifocal view: either in *focus* space or in *context* space. Interaction at focus scale allows accurate navigation but is slow on long distances. On the contrary, interaction at context scale allows fast navigation but is not accurate because of the magnification factor: a movement of one pixel at context scale corresponds to a movement of *mf* pixels at focus scale, where *mf* is the magnification factor.

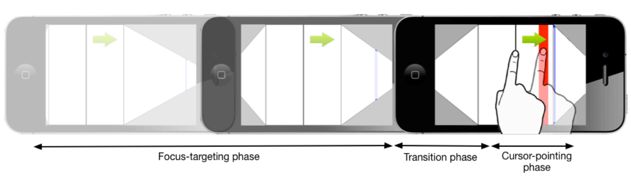


Figure 2. Selection task modeled as three phases.

For bifocal views, we therefore discard the modalities that rely on interacting solely in *focus* space (e.g., standard flick-drag) or that rely on interacting solely in *context* space. And we adopt a multimodal approach to solve the problem. We combine two modalities: the first one controlling the position of the focus area in *context* space to move quickly the cursor near the point of interest, and the second one controlling the position in *focus* space to move accurately the point of interest in the middle of the focus area. We investigate the complementary use (sequential/concomitant use) of these two modalities as described by the multimodal model of Serrano&Nigay [26]. To do so, we model the task as three phases (Figure 2): the fast navigation phase (named focus-targeting phase hereafter as Appert et al. [2], also named motion by Gutwin [12]), the accurate navigation phase (named cursor-pointing hereafter as Appert et al. [2], also named acquisition by Gutwin [12]), and also the modality switch (named transition phase hereafter).

We fix the modality used for the cursor-pointing phase. It relies on the accurate flick-drag modality provided by the iOS SDK. It is well integrated in iOS devices and therefore is a well-established standard for navigation. This modality is active only in the focus area. We made this choice to reinforce the interaction metaphor: when interacting in the focus area, users move the dataset rather than the focus area. Therefore, data remains under the user finger. In return, the focus area appears moving in the opposite direction than the one of the performed gesture as explained by Gutwin [12]. But the small length of the movements performed during this phase minimizes this issue: indeed the focus area movements are almost not perceptible. This modality is used as the precise modality during the cursor-pointing phase for all the combined modalities. We explore the combination of this well-known precise modality with 6 modalities for the focus-targeting phase. The design rationale is based on the type of navigation, with 2 modalities per type, as shown in Table 1.

Table 1. Six modalities for the focus-targeting phase

	Discrete	Continuous
Sequential	DiscretePress, FastDrag	ContinuousPress, Tilt
Non-sequential	DirectTouch, Peephole	

4.1 FastDrag and DirectTouch

FastDrag and DirectTouch are two modalities based on touch interaction. FastDrag is based on drag gestures in *context* space. Thus, a displacement of 1 pixel on screen causes a displacement of the focus area of 1 pixel in *context* space, and therefore a displacement of *mf* pixels in *focus* space (*mf* being the magnification factor). This minimizes clutching caused by scrolling a large dataset. The two modalities FastDrag and flick-drag are based on drag gestures. We distinguish them by the number of fingers on screen: one finger dragging in the focus area performs scrolling in *focus* space (cursor-pointing phase), two fingers dragging anywhere on screen performs scrolling in *context* space (focus-targeting phase). Thus, users can easily switch from a modality to another. The direction of scrolling is the same for both modalities: drag one or two fingers to the left moves data to the left, and therefore the focus area to the right.

DirectTouch enables direct jump to a new position. The user selects a point (by a touch gesture) anywhere in the context area to move it in the focus area. This allows a very fast non-sequential navigation and avoids clutching.. When selecting a point, an animation is triggered helping the user to keep her/his spatial orientation. Because of the magnification factor between *focus* space and *context* space, and the perspective effect of the bifocal view, the modality is not precise. Since the DirectTouch modality is based on touch click, the transition to the flick-drag modality is very fast.

4.2 Tilt

The promising results of Tilt presented by Fitchett&Cockburn [10] motivated us to implement a Tilt modality. This allows users to tilt the device around the vertical axis of the plane defined by the device. Tilting the device on the left (respectively right) moves the focus area to the left (respectively right). The more the device is tilted, the faster the scrolling speed is in *context* space. Our implementation uses accelerometers and gyroscopes of the device. We applied the angle/speed conversion described by MacKenzie [21] using the tilt magnitude with the following equation:

$$tiltMagn = Math.sqrt(pitch * pitch + yaw * yaw)$$

We defined the speed as *tiltMagn * gain*, with *gain* equal to 75, as the results of MacKenzie [21] reveals that a tilt gain in range [50, 100] is optimal. This allows users to attain scrolling speeds of between -4500 and 4500 pixels per second with an angle between -60° and 60° around the initial position. This initial position is defined by the device position at the application launch. At any time the users can recalibrate this position by doing a double tap on screen. As done by Fitchett&Cockburn [10], we defined a “safe” area in which no scrolling is performed ([-6.5°, 6.5°] around the initial position). This stable region allows avoiding of undesired scrolling due to tremors or modality switch. Users have therefore to place the device back into this safe area to stop scrolling.

Figure 3. Pressure sensors prototype.

4.3 ContinuousPress and DiscretePress

ContinuousPress and DiscretePress are two modalities based on pressure sensors. Despite the fact that we wanted to use device built-in sensors, we believe that these kind of sensors will be integrated in smartphones in a very near future (i.e. Apple 3D Touch technology), and good results were obtained by Spelmezan et al. [27] with these kind of sensors. We built a prototype using an Arduino Micro board and two pressure sensors Interlink Elektroniks FSR 400 as shown in Figure 3. Sensors were fixed under a phone shell. We added small pieces of fabric onto the phone shell to allow users to sense where to put their fingers without looking at the position. The Arduino board is fixed on the back of the phone. The left (respectively right) sensor moves the focus area to the left (respectively right). As the other modalities for the focus-targeting phase, the scrolling performs on *context* space.

With DiscretePress, sensors are used to perform discrete navigation. Then users navigate by clicking on sensors. As described by Spelmezan et al. [27], two levels of pressure trigger two different events: *light-click* and *strong-click*. *light-click* is triggered after a click with a minimum pressure of 0.5N and a maximum pressure of 2N. It performs a displacement of 80 pixels. *Strong-click* is triggered after a click with a minimum pressure of 2N. It performs a displacement of 800 pixels. *Strong-click* allows users to jump quickly toward the target and *light-click* allows them to be more precise in jumping near the target.

ContinuousPress uses pressure sensors to perform continuous navigation. The stronger the users press a sensor, the faster the scrolling speed is. The navigation remains active until the sensor is released. As we did not have an operational amplifier component to linearize pressure inputs (as done by Spelmezan et al. [27]), we defined 3 ranges of pressure with 3 different speed gains to simulate a linearization: 3 under 0.5N, 4 between 0.5N and 2.5N, and 6 over 2.5N. These factors are applied to values returned by the Arduino board that are numbers between 0 and 1023 (0 = no pressure, 1023 = max pressure). This allows having a scrolling speed similar to that of the Tilt modality.

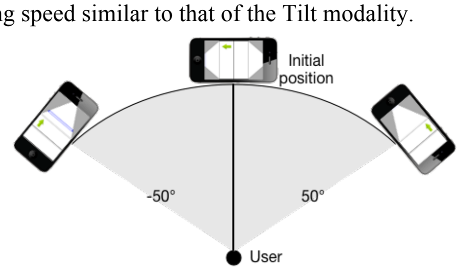


Figure 4. Peephole: a spatial-input-based modality.

4.4 Peephole

This modality uses spatial manipulation in order to map a physical position of the device to a virtual position in the dataset. This avoids interacting on the screen and therefore avoids clutching and occlusion problems. Three different mapping functions are presented by Pahud et al. [23]: a fixed planar mapping, a fixed spherical mapping and a dynamic mapping. We chose a spherical mapping (Figure 4). The spherical mapping is described by Pahud et al. [23] as difficult to use when spatial manipulation allows the users to zoom when moving the device closer to the body: this is not our case. Moreover a spherical mapping allows us to implement the modality without an accurate external positioning system (e.g., Optitrack): indeed we used the built-in compass to detect solely rotation around the user (using angle relative to the North). Finally users tend to sweep their arm in broad arcs when panning, as explained by Pahud et al. [23], so the spherical mapping seems to be a suitable mapping.

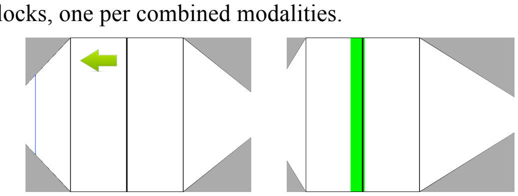
At the application launch, the current angle is retrieved and mapped to the middle of the dataset (because this is the initial position of the focus area). We restricted the movement amplitude in the range of [-50°, 50°] around the initial position (Figure 4). This allows the user to navigate in the entire dataset without making a large body rotation. This modality is activated on-demand, as described by Spindler et al. [28]. Activation is based on the right pressure sensor to avoid the occlusion problem with a finger on screen. The modality is active while the sensor is pressed; it is disabled when the sensor is released. This on-demand activation allows users switching from a relative to absolute mode and provides a stable position when the spatial navigation is disabled. In order to use spatial manipulation, users have to describe a rotation with their arms to the desired position and then press the sensor to move the focus area to the corresponding position.

5. EXPERIMENT

We conducted an exploratory controlled experiment to compare the performance of the six combined modalities described above. Participants were asked to perform pointing tasks with the implemented bifocal view. Our goal was to compare, for each combined modality the execution time for: the entire task, the focus-targeting phase, the cursor-pointing phase, and the transition between modalities. We therefore wanted to study each phase of the task depending on the combined modality.

5.1 Participants

We conducted an experiment with eighteen unpaid volunteers, 15 males and 3 females, aged 25 to 42 year-old (average 30.55, median 29), 14 were non-academic engineers in computer science, and 4 academic researchers in computer science. All of them were regular users of tactile mobile devices but had no experience with modalities except for flick-drag. The experiment was divided into six blocks, one per combined modalities.



(a) (b)

Figure 5. Task: (a) Beginning of the task: The arrow indicates the direction to the target. (b) End of the task: The cursor is within the target.

5.2 Method

Apparatus. We conducted the experiment on an iPhone 4 (960x640 pixels in landscape mode). The program was fully iOS coded, except a C program that gets pressure sensors values and sends them to the phone using WiFi.

Display. As explained before, in bifocal visualizations, we distinguish the focus space from the context space that are represented on screen at two different scales, respectively in the focus view and the context view (i.e., the two perspective sideviews). First in order to match with the implemented transformation function described by Leung&Apperley [20] that defines how the original context space is mapped to the two perspective sideviews, the focus view and the context view had the same size on screen. Hence the focus view and the context view (divided into two areas on screen) were 480 screen-pixels wide on iPhone 4. Second by assuming one value of the dataset per pixel, the size of the dataset is expressed in pixels, namely value-pixels. The selected size of *focus* space was 57 600 value-pixels, for enabling long distances to targets. Applying the value used in the Perspective Wall implementation of Mackinlay et al. [22], the size of *context* space was therefore 4800 value-pixels wide. We then had a factor (*mf*) equal to 12. We note that those parameters match with concrete application cases: for instance the visualization of a 10 year stock chart with a one-month period displayed in the focus: 480 pixels * 10 years * 12 months = 57600 pixels.

Task. We used the experimental protocol and task described by Appert et al. [2] for navigation of large datasets. Participants had to select targets in the context view. The cursor was displayed at the center of the focus view (Figure 5). Therefore the participants had to move the focus view to bring the cursor on the target and maintain it for 1 second. After that, a new target appeared. The cursor was presented as a 6 pixel wide black line (Figure 5) in order to be clearly visible on the screen. The target was presented as a 60 pixel wide red area (10 times larger than the cursor) in the focus view (Figure 5-a). So its size in the context view was $\frac{60}{mf}$, where *mf* is the magnification factor between *focus* space and *context* space. Then, the target is a 5 pixel wide in the context view (*mf* being equal to 12). The target size has been chosen to be easily selected because the concrete task is to move interesting data points into the focus view in order to obtain their details (i.e., the concrete task is not an accurate pointing task). We therefore do not want the difficulty of the task to be the accuracy of the selection during the cursor-pointing phase. Finally the target became green when the cursor is on it (Figure 5-b). Because of the perspective effect and the magnification factor, the target in the context view was not clearly visible. First a green arrow displayed at the top of the focus view indicated the direction of the target to minimize desert fog effects described by Jul&Furnas [17] (Figure 5-a). Second, the target was surrounded by a blue rectangle in the context view to be always perceptible. This avoids the stage of searching the target, which can be found by chance. We therefore focus only on navigation performance. The order of appearance of the targets forced participants to perform the task in left and right directions.

In our experiment while the magnification factor is defined by the bifocal visualization, we fixed the target size and varied the difficulty of the tasks by considering 4 distances for the target: D1=4800 pixels, D2=9600 pixels, D3=19200 pixels and D4=38400 pixels. Those distances are expressed in *focus* space (respectively 400, 800, 1600 and 3200 pixels in *context* space).

Procedure. We grouped trials into six blocks, one per combined modalities. After a brief demonstration of the combined modality, participants performed six training tasks before starting a block. They could ask any question during this step. The presentation order of the combined modalities was counterbalanced using a Latin square. For each block (i.e., each combined modality), participants had to perform 6 trials per distance. Therefore, they had to perform 24 (6 trials x 4 distances) tasks per combined modality and 144 selection tasks in total. The participants also answered a System Usability Scale (SUS) [6] on the combined modality they just experienced before starting the next one in order to collect the participant's subjective point of view on the modality s/he just experimented. The participants were instructed to be as fast and as accurate as possible. Finally, the participants were asked to rank the three combined modalities they preferred in order of preference and to explain their choices. The entire experiment lasted approximately 35 minutes.

5.3 Hypothesis

Based on the design rationale of the combined modalities, we formulate two hypotheses:

H1: Combined modalities providing direct access (DirectTouch and Peephole) will be faster during the focus-targeting phase than the modalities requiring navigation. The two modalities DirectTouch and Peephole are very inaccurate because of the magnification factor; therefore we also expect that the cursor-pointing phase will be longer because of the large remaining distance to be performed to reach the target.

H2: Combined modalities using the same modalities for the focus-targeting phase and the cursor-pointing phase (FastDrag, DirectTouch) will have a shorter transition time by reducing "homing" time.

5.4 Results and Discussion

5.4.1 Performance: execution time

For each task, we logged events (tactile screen, pressure sensors and embedded sensors) triggered by the participants. A post analysis of the generated log files allows us to measure accurately which phase is performed and for how long.

We check the normality of our data using the Shapiro-Wilk test. It reveals a deviation from the normal distribution, even with a transformation (log, square root). We then use the non-parametric Kruskal-Wallis test in order to test the significance of each factor (modality and distance) effect on each dependent variable (execution time for each phase). Focus-targeting execution time

The Kruskal-Wallis test reveals a significant effect for combined modalities on focus-targeting execution time ($p < 0.001$). A post-hoc test using Dunn with Bonferroni correction shows that there are significant differences between all pairs of combined modalities except between ContinuousPress and DiscretePress ($p = 0.1779$), between ContinuousPress and FastDrag ($p = 0.8912$) and between DiscretePress and FastDrag ($p = 0.2867$).

As expected (H1), DirectTouch is clearly the fastest modality (Figure 6-a), because it provides a direct access. However, for the Peephole modality, which also enables direct access to an area of the context by moving the device, the result is different. So H1 is not verified for Peephole. It is explained by the fact that participants used this combined modality in a continuous way for navigation. Therefore the participants spent more time than expected. This observed usage of the Peephole modality is in contradiction with our design rationale of providing a direct access by positioning the device according to a virtual 1D space in front of the user. Two reasons may explain this observed usage of the modality. First the interaction metaphor of moving the device that provides a window on a virtual information space (namely Lens-in-Hand metaphor by Pahud et al. [23]) is maybe stronger with a 2D virtual space as in [23, 28]. Moreover this Lens-in-Hand interaction metaphor is maybe perceived in opposition to the graphical metaphor of displaying the complete information space (focus and context) on the mobile device screen. For the rest of the analysis we consider that the Peephole modality is a continuous navigation modality as the Tilt modality for instance.

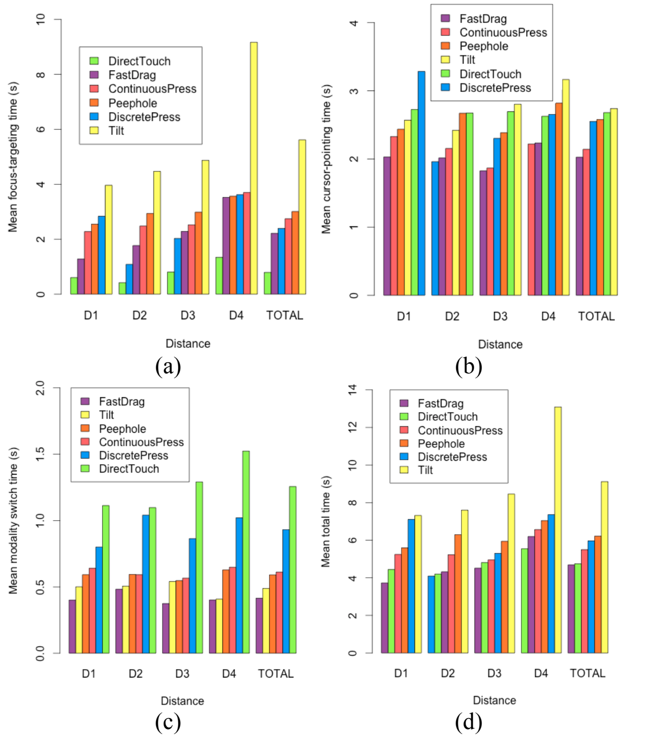


Figure 6. (a) Focus-targeting mean time (b) Cursor-pointing mean time (c) Transition mean time (d) Total mean time.

Execution time is much slower for Tilt. First we observed for the Tilt modality only that the participants accidently came back to the focus-targeting phase (i.e. tilt scrolling) after the cursor-pointing phase (i.e. touch gesture). Although we defined a "safe" area for the tilt ([-6.5°, 6.5°] around the initial position) in which no scrolling is performed, touch interaction provoked tilting the device outside the "safe" area and activated undesired scrolling. This observation explains that the focus-targeting execution time for Tilt was very long. Another explanation is the way participants used the combined modalities. Indeed we observed an unexpected navigation strategy. Rather than quickly moving the focus view not too far from the target to then switch to the precise modality, participants spent time trying to get the target displayed in the focus view. This strategy is perhaps encouraged by the fact that the precise modality implies finger touch gestures in the focus view only. Based on this strategy, the participants wasted time trying to be precise. So the execution time during this phase is related to the accuracy of the modality. This is confirmed by the good performance of the FastDrag modality, which is a precise one. Moreover clutching with FastDrag was minimized, making the modality even faster. With ContinuousPress, a light short press on one pressure sensor performs a very short navigation step and therefore enables the participants to be precise. Regarding Peephole, the on-demand activation with a press on the pressure sensor improves the accuracy. Participants have to simply release the sensor to stop the navigation. Whereas with Tilt, they have to come back to the static position and then anticipate the movement or navigate slower. This is why performance is better with Peephole than Tilt.

Finally, as shown in Figure 6-a, the execution time increases as the distance increases for all combined modalities, except for DiscretePress and DirectTouch. First DiscretePress is slower on the shortest distance. We explain this result by the difficulty for some participants to distinguish the two levels of pressure (confirmed by the user preference). Therefore they performed unwanted long movements. Second, despite that the DirectTouch provides a direct access, it is nevertheless dependent on the distance because of the perspective effect. Indeed the further the target is, the less visible it is and therefore easy to select. Therefore the task could require few more clicks, explaining why the execution time is slower for D4.

5.4.1.1 Cursor-pointing execution time

The Kruskal-Wallis test reveals a significant effect for combined modalities on cursor-pointing execution time ($p < 0.001$). The post-hoc test shows that there is no significant difference between ContinuousPress and DiscretePress ($p = 0.07909$), between ContinuousPress and FastDrag ($p = 0.4561$), between DirectTouch and Peephole ($p = 0.1267$), between DirectTouch and Tilt ($p = 0.5413$), and between Peephole and Tilt ($p = 0.5516$).

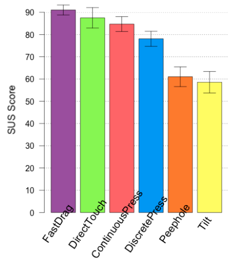
As expected, this phase is not dependent on the distance factor (Figure 6-b). Actually, the performance of this phase is dependent on the results of the focus-targeting phase, as explained by Appert et al. [2]. FastDrag is the fastest modality for this phase. This is due to the fact that this combined modality provides a better degree of control during the focus-targeting phase, so allows participants to position the target very close to the cursor. In the same way and as explained above, with the ContinuousPress, a light short press on the pressure sensor performs a very short navigation step, enabling the participants to be precise. Increasing the accuracy during the focus-targeting phase decreases the time of the cursor-pointing phase.

Nevertheless, since participants used the same navigation strategy for all combined modalities by trying to be precise during the focus-targeting phase, it is not surprising to observe small execution time variation between the modalities during this phase. So H1 is not verified: there is no significant difference between DirectTouch, Peephole and Tilt. We calculated the standard deviation for each phase on the entire dataset. We obtain 4.2 for the focus-targeting phase, 0.7 for the transition phase and 1.7 for the cursor-pointing phase. By computing the ratio with the mean time of each phase (2.45 seconds for the focus-targeting phase, 0.71 for the transition phase and 2.82 seconds for the cursor-pointing phase), the cursor-pointing phase has the smallest one (1.71 for the focus-targeting phase, 0.98 for the transition phase and 0.49 for the cursor-pointing phase). This means that the cursor-pointing phase variability is less important than for the two other phases, and that the total execution time with this navigation strategy is mainly influenced by the focus-targeting time and the transition time.

5.4.1.2 Transition time

The Kruskal-Wallis test reveals a significant effect for combined modalities on transition time ($p < 0.001$). The post-hoc test shows that there are significant differences between all pairs of combined modalities except between ContinuousPress and Peephole ($p = 0.6315$).

As expected, the transition time is not dependent on the distance of the target. FastDrag has the shortest transition phase (Figure 6-c), confirming H2 for this modality (no "homing" time). Since Tilt does not have to be activated, the transition can be anticipated, making it very fast: simply tilting the device for fast navigation and touching the screen for precise navigation. ContinuousPress and Peephole have a longer transition phase with similar transition times. For both modalities, we observed that several participants do not feel comfortable interacting on screen with the thumb. Therefore they have to remove their forefingers from the pressure sensors to perform the cursor-pointing phase, increasing the transition time.



(a)

Technique	#1	#2	#3	Score
DirectTouch	13	3	0	93
FastDrag	1	9	7	79
ContinuousPress	4	5	4	65
DiscretePress	0	0	4	16
Tilt	0	1	2	13
Peephole	0	0	1	4

(b)

Figure 7. (a) Mean SUS score with 95% confidence intervals (b) Preference ranking scores and the number of times the combined modalities were ranked 1st, 2nd or 3rd.

Finally Figure 6-c shows that DirectTouch and DiscretePress have a much longer transition phase. H2 is therefore not verified for DirectTouch. These two modalities involve a discrete mode of navigation control. This implies disorientation due to the loss of control of the movements, even though an animation was provided. The long transition is mainly due to this disorientation effect. This result shows that the articulatory switch (present for Tilt, Peephole, ContinuousPress and DiscretePress modalities) is less important than the disorientation issue caused by the loss of control of the movements with the two modalities DirectTouch and DiscretePress.

5.4.1.3 Total execution time

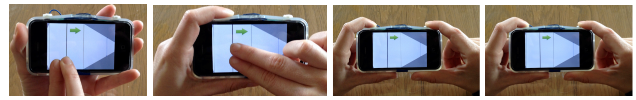
The Kruskal-Wallis test reveals a significant effect for combined modalities on total execution time ($p < 0.001$). The post-hoc test shows that there are significant differences between all pairs of combined modalities except between FastDrag and ContinuousPress ($p = 0.0734$) and between FastDrag and DirectTouch ($p = 0.2193$). The total execution times confirm the 3-phase modeling of the task. Figure 6-d shows that the three fastest combined modalities are DirectTouch, FastDrag and ContinuousPress. Despite that DirectTouch is very fast during the focus-targeting phase, the difference with the other modalities is seriously reduced for the total execution time, due to the long transition phase.Near-constant times observed during the cursor-pointing phase, show that this phase has little influence on the total execution time. We explained it by the navigation strategy adopted by the participants, trying to obtain the target displayed in the focus view before switching to the cursor-moving phase. With this strategy, the two main factors are the focus-targeting phase and the transition phase. Regarding the dependency between execution time and distance, tendencies are quite similar as those in the focus-pointing phase, reinforcing the fact that the total execution time strongly depends on the focus-targeting time and the transition time.

5.4.2 User Preference

User preferences reflect the performance results. In Figure 7, the modalities are ordered from left to right with decreasing order of mean SUS score. SUS questionnaire shows that participants ranked the FastDrag combined modality as the most usable one.

The preference ranking (Figure 7) is similar to the mean SUS score. The three preferred combined modalities have also been described as the most usable in the SUS questionnaire. We computed the ranking score (S) of a combined modality using the formula $S = 6*1^{st} + 5*2^{nd} + 4*3^{rd}$, where 1^{st}, 2^{nd} and 3^{rd} were its ranking at the corresponding place. We checked that with lower coefficients (respectively 3, 2, 1), the results remained similar.

Participants found the DirectTouch modality very easy to use: "After 1 or 2 clicks I know that I am close enough to switch to the precise modality". Only few participants reported the disorientation issue noted in the performance analysis. We think that the impression of ease and efficiency provided by this known modality compensated this problem.



(a) (b) (c) (d)

Figure 8. FastDrag (a) from bottom of the screen: limited occlusion (b) from the side of the screen: large occlusion. DiscretePress (c) Thumbs under the device: light-click and strong-click difficult to distinguish (d) Thumbs on the screen: strong-click hard to perform.

The FastDrag has been appreciated because it is a well-known modality. Even if clutching was minimized (movements in *context* space implying 4 or 5 drag gestures to reach very far targets), several participants reported the clutching issue. Moreover some participants also mentioned the occlusion problem. Depending on how the participants were holding the device, the fingers may occlude a large part of the screen (Figure 8).

Participants explained the bad scores for Tilt, DiscretePress and Peephole. For these combined modalities, several participants told us that they were too complicated for the simple task to be performed. Few of them said that in a real context usage, this perceived difficulty could prevent them to stay focused on the dataset.

- Tilt has been perceived as too sensitive. First in our implementation, we wanted to allow fast navigation without a large tilt angle and thus without reducing the screen visibility. However, some participants told us that the screen visibility while navigating was an issue. Second as explained above, participants tried to be as precise with this modality as with other modalities (e.g., FastDrag or DiscretePress) that allow more precise movements. Finally, we occasionally observed undesired tilt scrolling when the participants were performing touch gestures on screen.
- The DiscretePress modality has two problems according to participants. First and unlike DirectTouch, participants clearly expressed that there was a disorientation issue. They were not able to anticipate the position of the cursor after pressing a pressure sensor. This issue could be minimized and even disappear after a longer learning period. Secondly, depending on how participants were holding the device, they found it difficult to differentiate the light-click event from the strong-click event, or to trigger a strong-click event (Figure 8). Some participants suggested having a calibration step to allow them to choose their preferred levels of pressure.
- For the Peephole modality, almost all participants reported that the physical movements were too constraining and physically tiring. Moreover some participants explicitly mentioned the issue of the social acceptability of this spatial-input-based modality. Despite the fact that the Peephole modality is more commonly used with concave visualizations, and that bifocal view is a convex visualization, participants did not comment on this issue. Instead, they clearly identified the two metaphors: moving a lens on a ribbon (Peephole modality) and scrolling the ribbon (other modalities).

6. SUMMARY AND FUTURE WORK

With a bifocal view, tasks like moving an interesting data point into the focus area becomes difficult because of the large difference of scales between the context view and the focus view. Thus, interacting with a bifocal view requires both a fast access to data in the context view and high precision interaction with data in the detailed focus view. To address this issue, we have investigated a multimodal approach combining one modality for fast access to data with a precise modality for interacting in the focus view. We compared the combination of the well-known flick-drag technique (precise interaction in *focus* space) with six different modalities (fast access in *context* space). There are two main findings from the experimental study that can be used in designing combined modalities: (1) A task is decomposed into three distinct phases: a focus-targeting phase, a transition phase and a cursor-pointing phase. The observed navigation strategy is expressed by this 3-phase modeling of the entire task. (2) The articulatory switch between modalities less affects performance than the disorientation effect of some modalities caused by the loss of movement control.

As future work, the good performance and subjective preferences of the combined continuous modality based on pressure sensors incite us to further investigate this modality: fine-tuning the pressure-dependent control, study an automatic calibration phase according to how the user holds the device and study the position of the sensors to minimize the transition time between modalities.

Moreover we focus on interaction tasks only and more realistic reading or analysis tasks must be considered in further studies. To do so we plan to study the efficiency of the combined modalities for different types of tasks. Based on the taxonomy of Andrienko [1], we would like to consider synoptic tasks requiring us to explore the entire set of data: for instance find the maximum price of a company's stock. Another type of task can be the direct comparison between two distant data points. A general purpose combined modality for interacting with the bifocal view would ideally be efficient for all types of tasks.

7. REFERENCES

[1] Andrienko, N., Andrienko, G. *Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach*. Springer, 2005.

[2] Appert, C., Chapuis, O., Pietriga, E. High-Precision Magnication Lenses. *In Proc. of CHI 2010*. ACM, 273-282.

[3] Appert, C., Fekete, J-D. OrthoZoom scroller: 1D multi-scale navigation. *In Proc. of CHI 2006*. ACM, 21-30.

[4] Baglioni, M., Malacria, S., Lecolinet, E., Guiard, Y. Flick-and-Brake: Finger Control over Inertial/Sustained Scroll Motion. *In Proc. of CHI EA 2011*. ACM, 2281-2286.

[5] Bederson, B. B., Clamage, A., Czerwinski, M. P., Robertson, G. G. (2004). DateLens: A Fisheye Calendar Interface for PDAs. *ACM TOCHI*, 11, 1 (2004), 90-119.

[6] Brooke, J. SUS: A Quick and Dirty Usability Scale. *In Usability Evaluation in Industry*, (1996). Taylor&Francis, 189-194.

[7] Burigat, S., Chittaro, L. On the effectiveness of Overview+Detail visualization on mobile devices. *Springer Journal PUC*, 17, 2 (2013), 371-385.

[8] Büring, T., Gerken, J., Reiterer, H. User Interaction with Scatterplots on Small Screens - A Comparative Evaluation of Geometric-Semantic Zoom and Fisheye Distortion. *IEEE TVCG*, 12, 5 (2006), 829-836.

[9] Chittaro, L. Visualizing Information on Mobile Devices. *IEEE Computer*, 39, 3 (2006), 40-45.

[10] Fitchett, S. Cockburn, A. Evaluating reading and analysis tasks on mobile devices: a case study of tilt and flick scrolling. *In Proc. of OZCHI 2009*. ACM, 225-232.

[11] Ganhör, R. Athmos: Focus+Context for Browsing in Mobile Thumbnail Collections. *In Proc. of ICMR 2014*. ACM, 65-72.

[12] Gutwin, C. Improving focus targeting in interactive fisheye views. *In Proc. of CHI 2002*. ACM, 267–274.

[13] Gutwin, C., Fedak, C. Interacting with big interfaces on small screens: a comparison of fisheye, zoom, and panning techniques. *In Proc. of GI 2004*. CHCCS, 145-152.

[14] Huot, S., Lecolinet, E. Focus+context visualization techniques for displaying large lists with multiple points of interest on small tactile screens. *In Proc. of INTERACT 2007*. Springer, 219-233.

[15] Igarashi, T., Hinckley, K. Speed-dependent automatic zooming for browsing large documents. *In Proc. of UIST 2000*. ACM, 139-148.

[16] Jakobsen R. M., Hornbæk K. Sizing up visualizations: effects of display size in focus+context, overview+detail, and zooming interfaces. *In Proc. of CHI 2011*. ACM, 1451-1460.

[17] Jul, S., Furnas, W. Critical zones in desert fog: aids to mutiscale navigation. *In Proc. of UIST 1998*. ACM, 97-106.

[18] Kincaid, R. SignalLens: Focus+Context Applied to Electronic Time Series. *IEEE TVCG,* 16, 6 (2010), 900-907.

[19] Kratz, K., Brodien, I., Rohs, M. Semi-automatic zooming for mobile map navigation. *In Proc. of MobileHCI 2010*. ACM, 63-72.

[20] Leung Y.K, Apperley M.D. A review and taxonomy of distortion-oriented presentation techniques. *ACM TOCHI*, 1, 2 (1994), 126-160.

[21] MacKenzie, I.S., Teather, R.R. FittsTilt: The application of Fitts' law to tilt-based interaction. *In Proc. of* NordiCHI 2012. ACM, 568-577.

[22] Mackinlay, J.D., Robertson, G.G., Card, S.K. The perspective wall: detail and context smoothly integrated. *In Proc. of CHI 1991*. ACM, 173-176.

[23] Pahud, M., Hinckley, K., Iqbal, S., Sellen, A., Buxton, B. Toward compound navigation tasks on mobiles via spatial manipulation. *In Proc. of MobileHCI 2013*. ACM, 113-122.

[24] Quinn, P., Malacria, S., Cockburn, A. Touch Scrolling Transfer Functions. *In Proc of UIST 2013*. ACM, 61-70.

[25] Robertson, G.G., Mackinlay, J.D. The document lens. *In Proc. of UIST 1993*. ACM, 101-108.

[26] Serrano, M., Nigay, L. Temporal Aspects of CARE-based Multimodal Fusion. *Proc. of ICMI 2009*. ACM. 177-184

[27] Spelmezan, D., Appert, C., Chapuis, O., Pietriga, E. Side Pressure for Bidirectional Navigation on Small Devices. *In Proc. of MobileHCI 2013*. ACM, 11-20.

[28] Spindler, M., Schuessler, M., Martsch, M., Dachselt, R. Pinch-drag-flick vs. spatial input: rethinking zoom & pan on mobile displays. *In Proc. of CHI 2014*. ACM, 1113-1122.

[29] Wang, Y-S., Chi, M-T. Focus+Context Metro Maps. *IEEE TVCG,* 17, 12 (2011), 2528-2535.