SpaceFold and PhysicLenses: Simultaneous Multifocus Navigation on Touch Surfaces

Simon Butscher[1] Kasper Hornbæk[2] Harald Reiterer[1]

[1]Human-Computer Interaction Group, University of Konstanz, Germany

[2]Department of Computer Science, University of Copenhagen, Denmark

{simon.butscher, harald.reiterer} @uni-konstanz.de

kash@diku.dk

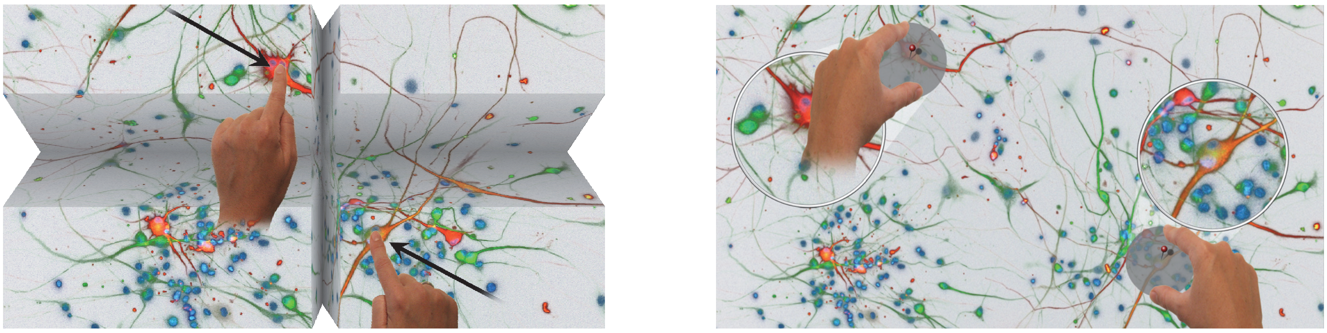

Figure 1: Comparing human neural stem cells: (left) SpaceFold – Fold the visual space like a sheet of paper to bring two areas of interest closer to each other; (right) PhysicLenses – Create multiple magnification lenses to see detailed views of areas of interest.

ABSTRACT
Many tasks performed in multiscale visual spaces require the user to have several foci. Using bimanual interaction, multitouch devices can facilitate the simultaneous definition and exploration of several foci. However, multitouch is rarely used for multifocus navigation, and may limit the interaction to a sequential definition of areas of interest. We introduce two novel navigation techniques that combine multiple foci and bimanual touch, and therefore enable the isochronic definition of areas of interest, leading to simultaneous multifocus navigation. *SpaceFold* folds the visual space in the third dimension, allowing users to bring objects closer to each other. Our technique enables a direct, bimanual manipulation of a folded space and therefore provides high flexibility. *PhysicLenses* uses multiple magnification lenses to compare objects. Using a physics model, *PhysicLenses* introduces a general solution for the arrangement of multiple lenses within the viewport. We conducted a controlled experiment with 24 participants to compare the techniques with split screen. The results show that *SpaceFold* significantly outperformed all other techniques, whereas *PhysicLenses* was just as fast as split screen.

Categories and Subject Descriptors
H.5.2 [Information Interfaces and Presentation]: User Interfaces - Interaction styles

General Terms
Performance, Design, Experimentation, Human Factors

Keywords
Bimanual input; multifocus visualization; multiscale navigation

AVI '14, May 27 - 30, 2014, Como, Italy

Copyright 2014 ACM 978-1-4503-2775-6/14/05…$15.00.

http://dx.doi.org/10.1145/2598153.2598177

1. INTRODUCTION
Many tasks that have to be performed in multiscale visual information spaces require several foci. This is the case for map navigation (e.g., comparing two islands), text editing (e.g., updating one part of a document with text from another), or file handling (e.g., copying files from one folder to another). Whereas virtually all interfaces allow the user to change between foci over time (e.g., by navigating a map with pan/zoom gestures, scrolling in a document, or switching among folders), fewer interfaces allow the simultaneous presentation of multiple foci. Examples of such multifocus interaction techniques include multi-window systems [1, 15], split-screen interfaces [16, 20], and a variety of research prototypes [4, 6, 8, 12, 18].

Multifocus techniques have been shown to be effective for exploring 2D spaces [8] and can give a good awareness of the intervening context [4]. Yet, research on how these techniques can be used on multitouch surfaces is limited. In contrast to many other input devices, multitouch surfaces seem to naturally enable multifocus interaction, for instance by using one hand per focus point. Previous research have merely used multitouch to sequentially define areas of interest [6, 9]. However, a bimanual approach for multifocus navigation would also enable users to simultaneously define areas of interest, which might increase users' efficiency and require fewer shifts between foci.

This paper presents two multifocus interaction techniques, called *SpaceFold* and *PhysicLenses,* both of which enable a simultaneous definition of the areas of interest. *SpaceFold* allows users to both pan/zoom and fold the visual space. A direct manipulation of the folds facilitated by multitouch interaction allows for a very high flexibility. *PhysicLenses* uses movable lenses to show magnified views of areas of interest defined by the user and introduces a generalizable way of automatically arranging several lenses within the visible viewport. Using a physical model of collisions, friction, and flexible connections, the arrangement of the lenses is controlled in a way that no overlapping occurs and at the same time pan/zoom interaction on the visual space is still enabled.

We describe the techniques and an experiment comparing the techniques with the baseline technique split screen. The results are twofold: First, we report on the performance and acceptance of *SpaceFold* and *PhysicLenses;* second, we give insights about the different characteristics of the techniques and how they influence the simultaneous definition of areas of interest. We used our results to compose five lessons for practitioners, which can be used to design techniques for simultaneous multifocus navigation on touch surfaces.

2. RELATED WORK

Multifocus techniques are used for many different tasks. One of them is comparing or relating parts of data. Such tasks recur in information visualization [19], for instance, and because only one graphical object can be held in memory [17], multifocus techniques may support visual comparisons of complex data that singlefocus techniques do not. Therefore, we first review multifocus techniques and then discuss how to use touch for multifocus interaction.

2.1 Multifocus Interaction

Providing several foci allows the user to simultaneously view areas of interest. Instead of time-multiplexing views (such as in zoomable user interfaces [2], and in focus+context techniques [12, 14]), multifocus is about space multiplexing. Using multiple windows to do so is a classic solution [1, 15]. LifeLines, for instance, is a multi-window interface which uses a split screen to explore large time series data [16]. Another solution divides the screen automatically into two viewports when two interaction points move apart [20]. Split screens, however, often provide little guidance as to the spatial relationships between viewports. PolyZoom [8] tries to overcome this problem by allowing the user to progressively build hierarchies of focus regions.

Focus+context techniques can also be extended to multifocus. For instance, some variants allow several fisheye lenses in a map-like environment [6, 20]. Another technique is Rubber Sheet [18], which allows the user to stretch or squish rectilinear focus areas. Instead of squishing the visual space, Mélange [4] folds it. The idea of folds is to give clearer information on the distance between the focus areas. However, Mélange is merely a presentation technique and the authors do not state how to interact with Mélange [4]. In an evaluation, the interaction was limited to a sequential definition of areas of interest through the positioning of one focus point with the help of a mouse. Furthermore, only one-dimensional folding was enabled. Another application of folded spaces is Canyon [7] which folds the visual space to visualize moving objects which are situated outside the focus area. Canyon only tracks predefined moving objects and does not address the definition of focus regions.

Folded representations of visual spaces appeared to have potential but the interaction with such folded spaces in terms of a simultaneous multifocus navigation has not been investigated yet.

While the techniques presented highlight multifocus navigation, none of them makes use of bimanual, multipoint-based input which is enabled by multitouch surfaces.

2.2 Multifocus Interaction with Touch

Multifocus interaction seems particularly well suited to touch interaction. Using bimanual, multipoint-based input, multitouch surfaces can facilitate the simultaneous definition of areas of interest. Nonetheless, little work has been done to explore this combination. Some papers have coupled touch with multiscale navigation (e.g., [3, 21]), but only few with multifocus. Four techniques stand out. In DTLens [6], users can create multiple lenses on a tabletop to interact with spatial data. It is not clear, however, whether it improves performance. I-Loupe [22] is a lens based interface and allows definition of a focus region and a magnified visualization of this area. Although not addressed by the authors it would conceptually be possible to use more than one i-Loupe simultaneously. However this technique does not provide solutions for problems like overlap and arrangement of the different lenses. The third technique is FingerGlass [9]. FingerGlass does not support multiple areas of interest, but it supports two-handed interaction with visual contents on multitouch screens. The non-dominant hand specifies an area of interest with which the dominant hand can interact in a magnified view. A user study showed that FingerGlass is much faster than four comparison techniques for the translation of objects within a visual space. Due to the singlefocus visualization of FingerGlass, however, multifocus interaction tasks like comparison of objects are not facilitated. In order to make use of lenses for a simultaneous multifocus navigation, new possibilities for the interaction and especially for the arrangement of the lenses have to be found. The forth technique is a physical information cloth [12]. The information is placed on a soft flexible cloth which can be draped, pulled, stretched, and folded with multiple fingers and hands. With the help of this technique, several visualizations like fisheye views or a perspective wall can be approximated. While this technique enables many opportunities, no evaluation of its efficiency was conducted. Thus, the potential of touch combined with multifocus remains unexplored, particularly for the comparison of objects within a multiscale visual space. Although there are some techniques for defining multiple areas of interest, all of them let the user define one area of interest after the other. We are interested in the possibilities of defining multiple areas of interest simultaneously in order to increase efficiency in terms of time and workload.

3. MULTIFOCUS NAVIGATION

We developed two multifocus navigation techniques for multiscale visual spaces. Both techniques enable simultaneous definition of areas of interest and therefore differ from known interaction techniques for multifocus views which are limited to a sequential approach.

3.1 Design Goals

Using the comparison of objects as an example for multifocus interaction tasks, we established three design goals for simultaneous multifocus navigation:

G1 – Multifocus View: Providing the opportunity to define multiple areas of interest is essential for multifocus interaction tasks, especially when objects, in particular complex ones, have to be compared [17].

G2 – Compatibility with Standard Gestures: In addition to multifocus interaction tasks like comparing and translating objects often also single focus navigation tasks like searching objects have to be performed. Therefore, techniques to support multifocus interaction tasks should not conflict with standard gestures for navigating in multiscale visual spaces popularized in commercial products, such as “Drag” and “Pinch” gestures, in particular.

G3 – Simultaneous Definition of Areas of Interest: In order to increase the efficiency, it has to be possible to define two areas of interest simultaneously. Leganchuk et al. [10] show that simultaneous bimanual input (both symmetric and asymmetric) is faster than sequential singlepoint-based input for controlling position and size of objects. Multifocus navigation techniques on multitouch surfaces can use these findings to support simultaneous bimanual input and simultaneous definition of areas of interest.

3.2 SpaceFold

Our novel multiscale interaction technique *SpaceFold* is a combination of the pan/zoom navigation enhanced with a distortion-based visualization technique. *SpaceFold* uses a metaphor of a folded sheet of paper (see Figure 1, left). By folding the visual space into the third dimension, it is possible to visualize local details without losing the context.

SpaceFold is inspired by Mélange [4], as well as by the work of Chiu et al. [3]. In contrast to Mélange, *SpaceFold* focuses on the interaction, not on the visualization. Mélange is mainly a presentation technique and does not stipulate how the user interacts with it [4]. Furthermore, Elmqvist et al. [4] stated that interacting with a folded space can be fairly complex and unintuitive. With *SpaceFold*, we present a technique which better couples the interaction and presentation in order to simplify the interaction with folded spaces. Although Elmqvist et al. showed that Mélange enables good awareness of the intervening context, the interaction technique as well as the evaluation had some limitations. Elmqvist et al. only evaluated their folding technique in a 1D scenario. No empirical evaluation for 2D folding was conducted. Furthermore, the interaction with the presentation technique was limited to control one focus point with the help of a mouse. A second focus point stayed at a predefined position. The positions of the focus points defined the areas of interest and therefore the folding of the visual space. Hence, Mélange focused on the positioning of focus points as a mediator between the user and the folding. We choose another approach, as we do not control focus points but the folding itself. Therefore, we can provide a more direct way of interacting with folded spaces. Using multipoint-based input, multitouch surfaces can simplify the definition of folds.

Chiu et al. [3] investigated gesture models for folding a virtual sheet of paper. Although Chiu et al. compared different models, they limited themselves to unidirectional folding (either horizontal or vertical). Furthermore, they predefined the positions of the folds and only manipulated the angle parameter. They compared if an asymmetric parallel gesture outperforms an asymmetric serial gesture. However, the requirements for the comparison of objects within a folded space are completely different, as it is up to the user where to place a fold.

Therefore, we know that folding the visual space has some advantages over split screen regarding the visualization, but the knowledge about how to interact with such a visualization is limited. Our *SpaceFold* techniques used the idea of folding the visual space, but we are interested in how to bimanually interact with such folds on a multitouch device in order to define two areas of interest simultaneously.

Figure 2 shows how *SpaceFold* helps inspect two areas of interest (**G1**). With *SpaceFold*, the multiscale visual space can be navigated with pan/zoom interaction (**G2**). If two areas of interest in the visible viewport need to be compared, the user taps and holds the areas of interest for a short period of time (200 ms) (**G3**). This activates the fold mechanism and users get a preview of the fold they are going to create (see Figure 2.1). We use dwell time because multitouch gestures that use more fingers on each hand are imprecise and uncomfortable for some orientations. Although dwell time was short, it was sufficient to distinguish fold and zoom interaction. The fold is created by moving the fingers towards each other (see Figure 2.2). The user can now zoom in again to see both areas of interest in greater detail (see Figure 2.3). As for conventional zooming, the zoom target is in the center between the two touch points.

Create Folds: The user can create a horizontal, a vertical, or a horizontal and a vertical fold simultaneously. In contrast to Mélange [4, 5], we implemented a solution for regions where vertical and horizontal folds overlap each other. In such situations, we deviate from the real world behavior of a folded sheet of paper and draw a nonlinear space. By decreasing the distance between the touch points, the multiscale visual space is folded directly. The width of the fold decreases while the depth increases. The center of the fold is placed halfway between the fingers. This allows the users to perform the gestures either symmetrically or asymmetrically and therefore increases the degrees of freedom as well as the simplicity, as the users do not have to think about the correct execution of a specific gesture. The width of the fold corresponds to the distance between the fingers minus an offset to ensure that the areas around the touch points do not fall within the fold. We use the whole space between the two touch points for the fold because it corresponds to folding a sheet of paper. In addition, thanks to the larger fold, the content falling within the fold has a better perceptibility than it would have for a small fold with a huge depth. Additionally, a fold has a minimum allowable width to ensure the visibility of the fold; folding to a width of zero leads to an unfold animation to the minimum width when touch is released. The width of a fold on the screen is independent of the zoom level. Therefore, folds do not occupy much space on the screen. To represent the change in scale within the folds, the depth of the folds is adapted: Zooming into the multiscale space leads to an increase in depth while zooming out decreases depth (see Figure 2.3).

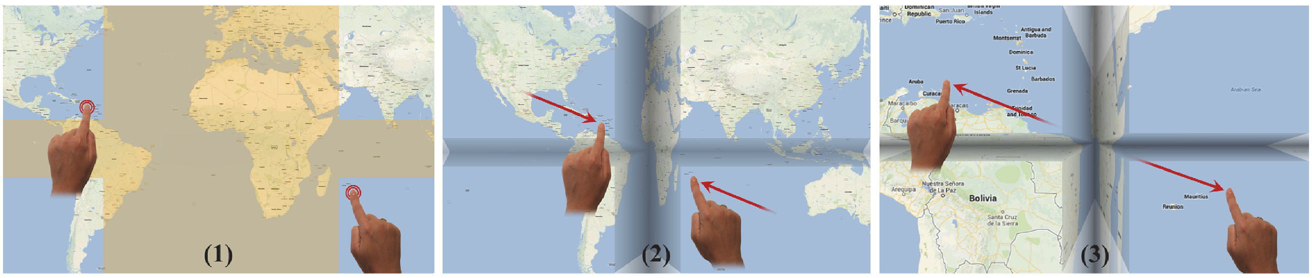

Figure 2: SpaceFold – Create fold to compare the sizes and number of islands in the Caribbean and Mauritius: (1) User taps on and holds the areas of interest and gets feedback on the fold position; (2) User moves fingers towards each other to fold the map; (3) User performs a simple zoom gesture to enlarge the areas of interest.

Modify Folds: *SpaceFold* supports multiple folds. If some folds already exist and the user wants compare another area, one of three things happens: (1) At least one of the areas is situated inside a fold, (2) the folds are placed between the areas, or (3) the areas are not affected by existing folds. For the latter case, the space can simply be folded as explained before. If an area of interest is situated inside a fold, the user is able to pull the area out of it. If there is a fold between the areas, the space has to be pushed into the fold. In both cases, the folds which are intended to be manipulated are selected by placing one finger on each side of the fold (see Figure 3 and Figure 4). Again, the folds are highlighted as a preview. If the user moves the fingers towards each other, the space is pushed into the existing fold (see Figure 3, vertical fold). If more folds are between the touch points, the space is first pushed into the folds next to the finger, then the distance between the folds decreases, and the folds are merged (see Figure 4). If the user moves the fingers away from each other, the space is pulled out of the fold (see Figure 3, horizontal fold).

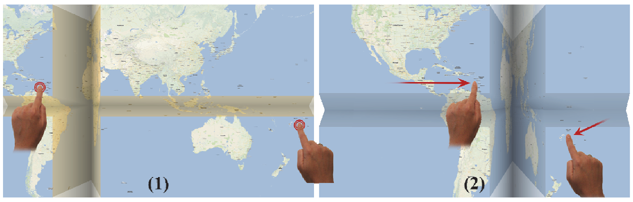

Figure 3: SpaceFold – Modify fold: User moves fingers towards each other on the horizontal axis to push the maps into the fold. At the same time, the user pulls some parts out of the fold by increasing the vertical distance between the fingers.

Horizontal and vertical folds can be manipulated simultaneously. Therefore, it is possible to merge two vertical folds while creating a horizontal fold, or to push space into a horizontal fold while merging two vertical folds. Users only have to know that if they grab the visual space on two areas of interest and move the fingers closer, the folds are modified and the two areas move closer to each other (see Figure 4).

Using bimanual multitouch for multifocus navigation has the advantage of manipulating and defining several areas of interest simultaneously. *SpaceFold* in the way it is shown here makes use of this like no other technique before. With *SpaceFold,* parameters like the zoom level or the visible viewport only have to be defined once for multiple areas of interest. This reduces the interaction steps which have to be performed and therefore simplifies the interaction. If necessary, different zoom levels for different areas of interest could be supported by a differentiation between pinch to zoom gestures within an area (zoom within one specific area) and gestures spanning multiple areas of interest (zoom whole visual space, see Figure 2.3). In contrast to Mélange, *SpaceFold* does not make use of focus points as mediators between the user and the folding. *SpaceFold* enables a direct manipulation of the folded space and therefore increases the flexibility. With *SpaceFold*, the users are free to define the screen distance between objects and therefore decide how important the intervening context is in a certain situation.

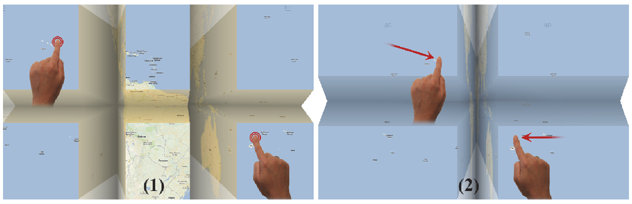

Figure 4: SpaceFold – Merge folds: User moves fingers towards each other to merge the existing two vertical folds and simultaneously pulls parts of the map into the horizontal fold.

3.3 PhysicLenses

PhysicLenses is another interaction technique that aims to fulfill the three design goals introduced. *PhysicLenses* enhances a pan/zoom interface with lenses showing magnified views of user-defined areas of interest (see Figure 1, right). *PhysicLenses* is inspired by multitouch lenses like FingerGlass [9]. In contrast to FingerGlass, *PhysicLenses* uses multiple foci (**G1**). Although, FingerGlass is optimized for the translation of objects, it is only a single focus technique and does not help in performing multifocus tasks like the comparison of objects. In contrast, *PhysicLenses* is optimized for the simultaneous creation of multiple lenses and the arrangement of these lenses within the viewport. *PhysicLenses* uses an approach similar to DragMag [23], where the magnified view of an area of interest is displayed with an offset to the original position. To control the arrangement of the lenses, *PhysicLenses* uses a physical model of collisions, friction, and flexible connections between the areas of interest and the magnification lenses. Using such a physics model, we introduce a general solution for the positioning of multiple lenses. In contrast to the other techniques mentioned, it is still possible to pan/zoom the multiscale visual space even when using *PhysicLenses* (**G2**). *PhysicLenses* can be used either with one hand or bimanually in order to define two areas of interest simultaneously (**G3**). Figure 5 shows a walkthrough of *PhysicLenses.*

Create Lens: To create a magnification lens, the user touches the screen with two fingers and holds them for a short period of time (200 ms). The distance between the touch points sets the diameter of the circular area of interest. An enlarged copy of the area of interest is immediately shown in a magnification lens (see Figure 5.1). The initial magnification ratio is three. We decided to use three as default magnification ratio as it seems a good tradeoff

Figure 5: PhysicLenses – Create lenses to compare the sizes and the number of islands in the Caribbean and the South Seas: (1) User taps with two fingers to define area of interest and therefore create a magnification lens; (2) User downsizes the area of interest and at the same time increases the magnification ratio in the magnification lens; (3) While panning the visual space, the magnification lens stays within the visible viewport; (4) User creates a second magnification lens to compare the two areas of interest.

between the magnification and the space needed. The user can adjust the area of interest by changing the distance between the fingers and the position of the hand. The magnification lens is updated in real-time. When changing the size of the area of interest, the size of the magnification lens stays at the size of creation time (three times the size of the area of interest), therefore the magnification ratio changes (see Figure 5.2). This mechanism enables the users to define the area of interest as well as the zoom level within the magnification lens in one continuous gesture.

Modify Lens: *PhysicLenses* integrates functionalities to manipulate the single lens parameters separately (see Table 1):

- *Area of Interest:* The area of interest can be resized by a two finger stretch and repositioned by a one finger drag. To preserve the relation between an area of interest and the corresponding magnification lens, the magnification lens is bound to the area of interest. Thus, if an area of interest is moved, the magnification lens follows.
- *Magnification Lens:* For fine adjustment, pan/zoom interaction is enabled within the magnification lens. Using a pinching gesture within the lens changes the magnification ratio. To preserve the clipping inside the lens, the magnification ratio and the size of the lens change. If the user touches the border of the lens with two fingers, the size of the lens can be manipulated without simultaneously changing the magnification ratio. To remove a lens, it has to be scaled down below a certain threshold.

Table 1: Effect of pinch and drag gestures on the different parts of the PhysicLenses interface

Gesture	Environment	Area of Int.	Lens (inside)	Lens (border)
Pinch	zoom	size	zoom	size
Drag	pan	move	pan	move

Arrange Lenses: A physical model handles the arrangement of the lenses. A detection of collisions and a handling of flexible joints between the areas of interest and the corresponding magnification lenses enable the use of multiple lenses simultaneously. The integration of flexible joint tries to keep the distance between the area of interest and the magnification lens low but at the same time allows the user to reposition the magnification lens manually. The collision detection avoids overlapping between the lenses and repositions them in a smooth transition if necessary. In order to guarantee the visibility of the magnification lenses, collisions with the viewport borders are also handled. The number of lenses is not limited, but as the screen is, at some point the lenses will overlap each other. Using multipoint-based input with the lenses in combination with a physics model, *PhysicLenses* introduces a general solution for the positioning of multiple lenses within the viewport.

4. EVALUATION

We conducted a study to gain insight into the characteristics of the techniques and how they influence the simultaneous definition of areas of interest. We were interested in how participants interacted with the techniques and focused on whether the participants used the techniques one-handedly or bimanually, whether they defined the areas of interest simultaneously or sequentially, and whether they performed the gestures symmetrically or asymmetrically. Furthermore, we were interested in the performance and acceptance of *SpaceFold* and *PhysicLenses*, as they are both novel multifocus interaction techniques. We used split screen as the baseline. In this condition, the screen was vertically separated into two viewports. Split screen is commonly used in practice and it also enables a simultaneous definition of areas of interest. Furthermore its interaction is simple. All three techniques used the same pan/zoom functionality.

4.1 Task

Participants were instructed to complete a comparison task as a kind of a multifocus interaction task. Two objects were shown on the multiscale space (see Figure 6) with a given distance (450 pixels to 900 pixels distance at 1:1 zoom level) and orientation (0°, 45°, 90°, 135°). The objects were positioned within an information landscape consisting of a dark gray square. Due to the lack of navigational cues, a red grid was used as background. Each object consisted of five colored rectangles (colors used were red, green, blue, black, yellow, purple, and cyan). The colored rectangles appeared at a certain zoom level (1:6 zoom level or above, which leads to a distance of at least 2700 pixels and 5400 pixels, respectively). The participants had to determine if the color codings matched each other. Users could then confirm whether or not the objects matched by activating one of two corresponding physical buzzers positioned on the left and right side of the display.

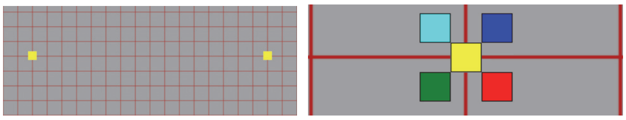

Figure 6: Objects to compare positioned on a gray square.

4.2 Design & Participants

The study used a within-subjects factorial design with three independent variables: *technique* (split screen, *SpaceFold*, and *PhysicLenses*), *orientation* of the objects (0°, 45°, 90°, 135°), and *distance* between the objects (450 pixels to 900 pixels). As the distance between the objects on the screen depends on the zoom level, we decided not to add the zoom level to the independent variables. As dependent variables, the completion time and the interaction steps performed (e.g., pan, zoom or fold actions) were logged. Furthermore, the subjective perceived workload (NASA-TLX) was measured.

Techniques were presented in a Latin squared order. For every technique, 4 blocks had to be completed. Each block contained a random order of the combinations of each distance with each orientation. We collected a total of 3 (*techniques*) x 4 (*blocks*) x 2 (*distances*) x 4 (*orientation*) = 96 trials from each participant.

We recruited 24 participants (12 female) for the study. Participants were between 19 and 57 years old with an average age of 26.3 years ($SD = 7.8$). Only one participant had no experience with smartphones, tablets, or larger multitouch devices. All participants were right-handed. The experiment took about an hour and participants were paid € 12 as compensation.

4.3 Apparatus

The experiment was done on a 27” Lenovo Horizon All-In-One capacitive multitouch display. The active display area is 597 x 336 mm in size and provides a resolution of 1920 x 1080 with a pixel size of 0.31 x 0.31 mm. The participants were free to choose a tilt angle for the display that was comfortable for them.

SpaceFold, *PhysicLenses*, and the baseline technique were implemented in Microsoft .NET 4.0 using C# and WPF. For the 3D functionalities of *SpaceFold*, we used the 3D APIs of WPF. For the automatic arrangement of the *PhysicLenses*, we integrated the Farseer Physics Engine (http://farseerphysics.codeplex.com/).

4.4 Hypotheses

With our two novel multifocus navigation techniques as well as with split screen, the areas of interest can be defined simultaneously and the gestures can be performed symmetrically or asymmetrically. We assume that the number of individual parameters to define an area of interest has an effect on the usage of the techniques. Techniques that couple the parameters defining the areas of interest (zoom factor and visible viewport) can help to reduce complexity. In contrast to *SpaceFold*, for *PhysicLenses* as well as for split screen the participants have to define these parameters for each area of interest individually. Therefore, although *PhysicLenses* and split screen enable a simultaneous definition of areas of interest, we expect that these will most often be used in a sequential way using only one hand (**H1**). Furthermore, we assume that a simultaneous definition of areas of interest outperforms sequential approaches in terms of time. Therefore, we expect that the task completion time for *SpaceFold* will be significantly lower than for the other techniques (**H2**).

5. RESULTS

We performed repeated measures analyses of variance to understand the effect of the technique on performance and self-report measures. The Greenhouse-Geisser adjustment was used for non-spherical data and the Bonferroni adjustment for post-hoc comparisons. Trials with a difference of more than three standard deviations from the mean were removed (31 trials, equivalent to 0.7% of the trials). To lower the influence of learning effects on task completion time, we removed the first block and aggregated the remaining three blocks for further analysis (there was no significant interaction between block and techniques).

5.1 Task Completion Time

Figure 7 shows the average task completion times across the techniques. We found a significant main effect for *technique*, $F(2,46) = 13.56$, $p < .005$, $\eta^2 = .37$. Pairwise comparisons showed that the average completion time for *SpaceFold* ($M = 7.19$s, $SD = 1.35$) was significantly lower than for *PhysicLenses* ($M = 8.84$s, $SD = 2.74$) and split screen ($M = 8.21$s, $SD = 2.05$). This means that task completion time for *SpaceFold* is 88 % the time for *PhysicLenses* (effect size of comparison is $\eta^2 = .57$) and 81 % the time for split screen ($\eta^2 = .38$). Task completion times for *PhysicLenses* and split screen were not significantly different. Note that the variability of performance with *SpaceFold* is much lower than for split screen and *PhysicLenses*.

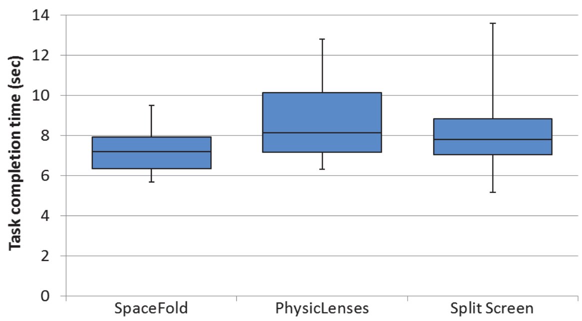

Figure 7: Performance metrics for comparison task.

Interaction data show that participants rarely adjusted the folds of *SpaceFold* (0.43% of the trials). With *SpaceFold,* participants performed an average of 4.77 interaction steps (pan, zoom, and fold; $SD = 1.59$). For *PhysicLenses*, participants performed 6.34 interaction steps on average ($SD = 2.54$). The high standard deviation reflects that participants interacted in quite varying ways. In 38% of the trials, participants were able to define the lenses with one gesture for each lens without the need for further adjustment. In other situations, the participants were not sure how to proceed and therefore tried different possibilities, leading to a high number of interaction steps. With split screen, participants used 9.29 pan and zoom actions to complete a trial ($SD = 3.77$). It seemed that participants lost their orientation in some situations and had to zoom out and look for the objects on a higher zoom level.

We found a number of interaction effects relating to distance and orientation (see Figure 8). First, we found an interaction between *technique* and *distance*, $F(1.36, 31.30) = 25.91$, $p < .005$, $\eta^2 = .53$. Post-hoc comparisons showed that *SpaceFold* was significantly faster than split screen and *PhysicLenses* for both distances (all $p < 0.05$). For the long distance, the speed-up of split screen in comparison to *PhysicLenses* was significant, but not for short distances. These data show that *PhysicLenses* was more sensitive with respect to distance. For short distances, the participants were able to adjust to a higher zoom level and see both objects when creating the lenses. Because of the higher initial zoom level, the participants were able to define the correct area of interest as well as the magnification ratio with one continuous gesture (average interaction steps $M = 5.87$, $SD = 2.34$). Because of the lower initial zoom level for large distances, a larger relative scale-up had to be performed. Therefore, the lenses had to be adjusted more frequently with regard to the size of the magnification lens and the magnification ratio (average interaction steps $M = 6.81$, $SD = 2.64$).

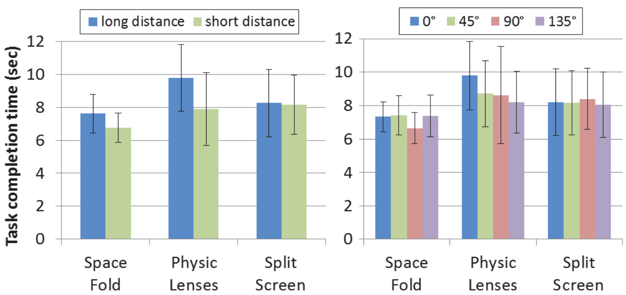

Figure 8: Average task completion times: (left) divided into distances; (right) divided into orientations.

Orientation was also interacting with *technique*, $F(3.49, 80.19) = 6.21$, $p < .005$, $\eta^2 = .21$. Post-hoc comparisons showed that *SpaceFold* outperformed split screen and *PhysicLenses* for all orientations, while split screen was significantly faster than *PhysicLenses* for an orientation of 0°. The performance of *PhysicLenses* dropped significantly for 0° compared with the other orientations. As already shown, lenses are very sensitive with regard to the distance between the objects. For an orientation of 0°, the screen has the shortest extension and therefore the maximal zoom level for which both objects are still visible is lower than for the other orientations.

5.2 Subjective Effort and Preference

The overall workload measured with NASA-TLX showed no significant differences between techniques. Also, the comparison of subscales showed no significant differences except for Physical Demand where the workload for *SpaceFold* and for *PhysicLenses* is significantly lower than for split screen ($p < 0.05$).

In addition to the NASA-TLX, participants rated the simplicity of the three techniques on a 5-point Likert scale ranging from -2 =

very complicated to 2 = very simple. According to a Wilcoxon signed-rank test there were no significant differences between *SpaceFold* (*Mdn* = 1), *PhysicLenses* (*Mdn* = 1) and split screen (*Mdn* = 0.5). We also asked participants which technique they preferred. *SpaceFold* was mentioned most often (10), followed by *PhysicLenses* (8), and finally the split screen interface (6).

With the help of semi-structured interviews, we identified perceived advantages and disadvantages of the techniques. *SpaceFold* was preferred because it felt the most natural, just like folding a paper map. One participant said that it felt like folding the world into the preferred shape. Some participants mentioned that *SpaceFold* requires only a few interaction steps and that they needed only select among few interaction mechanisms. The straightforward use of bimanual input was also emphasized. For *PhysicLenses*, the participants noted that many different functions have to be considered. Most of them recognized that *PhysicLenses* can be used very quickly once accustomed to it.

5.3 Gestures

Alongside the evaluation of the performance, a qualitative analysis of video recordings was conducted to investigate the gestures used. As participants were able to perform either symmetric or asymmetric gestures and define the areas of interest either simultaneously or sequentially, we were able to investigate the preferences of the users and on what characteristics of the techniques they depend. Although possible with all three techniques, the simultaneous definition of areas of interest was almost exclusively used for *SpaceFold*, where the simultaneous definition is by design. For *PhysicLenses*, two participants created two lenses simultaneously but nevertheless defined the individual parameters of the lenses in a sequential way. Only one participant used the *PhysicLenses* to simultaneously create and define two lenses with two hands. For split screen, only one participant chose a simultaneous approach. The participant first centered one object in each viewport and then zoomed into the viewports simultaneously using both hands. Fine adjustment was done in a sequential way again. The participant tried to perform the gestures in a way that she could move both hands symmetrically in order to reduce complexity.

Bimanual input strongly depended on the simultaneous or sequential definition of areas of interest as a simultaneous definition was only possible through bimanual interaction. Participants tend to stay in the mode they were in. Therefore, most participants (n=19) used two hands to zoom in when using *SpaceFold*, as they also folded the space with two hands. In contrast, for split screen (n=14) as well as for *PhysicLenses* (n=11), the number was a bit lower. Generally, we were able to observe that when participants started to interact with both hands, they tended to stay in this mode throughout the whole study regardless of which technique they were using.

Observations showed that participants preferred the use of symmetric gestures over asymmetric gestures. For all techniques, the participants used mainly symmetric gestures to e.g. zoom, fold, or resize lenses. However, five participants tended to conduct asymmetric gestures when folding the visual space.

6. DISCUSSION

The study confirmed that the simultaneous definition of areas of interest is less time consuming than a sequential definition. *SpaceFold* was the only technique the participants used to create the areas of interest simultaneously (**H1**). Furthermore, *SpaceFold* was significantly faster than the other techniques (**H2**). Next, we illustrate reasons for the differences identified and discuss ways in which the principles underlying our techniques can be used for future multifocus navigation techniques.

6.1 Explanation for Results

The main difference between the techniques was the number of parameters which have to be controlled for each area of interest. For *SpaceFold* the zoom level as well as the visible viewport had to be defined only once, but for *PhysicLenses* as well as for split screen the participants had to define these parameters for each area of interest separately. This showed to have an effect on the number of interaction steps necessary to complete the task as well as on the task completion time.

SpaceFold: *SpaceFold* couples the parameters to define the individual areas of interest (zoom level and visible viewport). Therefore, the definition of the areas of interest required fewer interaction steps than for the other techniques. Furthermore, *SpaceFold* showed to be robust against variations in orientation. For the participants, it made no difference whether they had to create a vertical fold (90°), a horizontal fold (180°), or both folds at the same time (45° and 135°). The subjective evaluation showed that participants found the interaction steps easy to remember and simple to perform. This is also substantiated by workload ratings: *SpaceFold* was rated better or as good as split screen for all subscales.

PhysicLenses: The participants were able to define the magnification ratio within the lens as well as the size and position of the magnification lens for each area of interest separately. This increase in flexibility has come at the expense of simplicity. The subjective evaluation showed that too many different functions had to be distinguished by the participants. Although it was possible to use *PhysicLenses* bimanually, most of the participants preferred to use their dominant hand and solved the task through a sequential definition of areas of interest. Therefore, in order to facilitate a simultaneous definition, the functions should be reduced or aggregated.

Split Screen: For split screen, each area of interest had to be defined separately. Therefore, the zoom level as well as the visible viewport had to be defined twice. Moreover, it is time consuming and participants stated that it was annoying to do the same task two times, which is also represented in the significantly higher rating for physical demand. Furthermore, we identified the same problems, such as loss of context and orientation or the tedious navigation, that other studies have revealed [4, 11, 13].

6.2 Lessons for Practitioners

We identified several findings which should be taken into consideration when designing interaction techniques for multifocus navigation on touch surfaces:

- Simultaneous definition of area of interest made a significant difference in our study. For devices enabling multipoint-based input, simultaneous definition should be facilitated.
- For simultaneous definitions of areas of interest, the number of parameters that need to be controlled individually for each area of interest (like the zoom level or the visible viewport) has an effect on the complexity. To enable a simultaneous definition, these parameters have to be coupled between the regions (e.g. same zoom level for all areas of interest).
- Symmetric gestures were preferred by the participants. To reduce complexity, users should not have to distinguish between symmetric and asymmetric gestures for triggering functionalities.

- The use of a dwell time to explicitly trigger functions works well. The observations during the study showed that the creation of folds and lenses did not conflict with the pan/zoom navigation, and unintended use of the function seldom occurred. As the users are very accustomed to the pinch to zoom gesture they use it very unconsciously. In contrast the additional functionality triggered by dwell time is used more intentionally and for that reason more slowly. Therefore, in our study the dwell time was rarely noticed by the participants. Furthermore, we observed that a very short dwell time (200 ms) is enough to ensure a robust differentiation. Therefore, dwell time is a good alternative to the use of asymmetric gestures (like for example recommended by Chiu et al. [3]) in order to distinguish additional functionality from a symmetric pinch gesture to zoom.
- The *SpaceFold* as well as the *PhysicLenses* technique can be adopted for many situations in order to support multifocus interaction tasks. Both techniques facilitate the awareness of the intermediate context. One advantage of *SpaceFold* in contrast to all other techniques is that the areas of interest always have the same zoom level. This facilitates comparisons in which the dimensions of objects have to be taken into account. Designers should carefully think about how many degrees of freedom (like e.g. individual zoom level for different areas of interest) are necessary and limit the number to a minimum.

7. CONCLUSION

In this paper, we investigated the usage of multifocus interaction techniques for bimanual multipoint-based input like it is enabled by multitouch surfaces. We presented two novel multifocus interaction techniques: *SpaceFold* and *PhysicLenses*. Both enable a simultaneous definition of areas of interest but differ in the coupling of parameters to define the areas of interest. An experimental comparison of the two novel techniques with a split screen interface showed that *SpaceFold*, the technique which was used to define areas of interest simultaneously significantly outperformed all other techniques. Based on our results, we defined five lessons for practitioners which should be taken into consideration when designing interaction techniques for simultaneous multifocus navigation on touch surfaces.

8. REFERENCES

[1] Baldonado, M.Q.W., Woodruff, A. and Kuchinsky, A. 2000. Guidelines for Using Multiple Views in Information Visualization. In Proc. *AVI'00*, ACM, 110–119.

[2] Bederson, B.B. and Hollan, J.D. 1994. Pad ++ : A Zooming Graphical Interface for Exploring Alternate Interface Physics. In Proc. *UIST'94*, ACM, 17–26.

[3] Chiu, P., Liao, C. and Chen, F. 2011. Multi-touch Document Folding: Gesture Models, Fold Directions and Symmetries. In Proc. *CHI'11*, ACM, 1591–1600.

[4] Elmqvist, N., Henry, N., Riche, Y. and Fekete, J.-D. 2008. Mélange: Space Folding for Multi-Focus Interaction. In Proc. *CHI'08*, ACM, 1333–1342.

[5] Elmqvist, N., Riche, Y., Henry-Riche, N. and Fekete, J.-D. 2010. Mélange: Space Folding for Visual Exploration. *IEEE Transactions on Visualization and Computer Graphics*. 16, 3 (2010), 468–483.

[6] Forlines, C. and Shen, C. 2005. DTLens: Multi-user Tabletop Spatial Data Exploration. In Proc. *UIST'05*, ACM, 119–122.

[7] Ion, A., Chang, B.Y.-L., Haller, M., Hancock, M. and Scott, S.D. 2013. Canyon: Providing Location Awareness of Multiple Moving Objects in a Detail View on Large Displays. In Proc. *CHI'13*, ACM, 3149–3158.

[8] Javed, W., Ghani, S. and Elmqvist, N. 2012. PolyZoom: Multiscale and Multifocus Exploration in 2D Visual Spaces. In Proc. *CHI'12*, ACM, 287–296.

[9] Käser, D., Agrawala, M. and Pauly, M. 2011. FingerGlass: Efficient Multiscale Interaction on Multitouch Screens. In Proc. *CHI'11*, ACM, 1601–1610.

[10] Leganchuk, A., Zhai, S. and Buxton, W. 1999. Manual and Cognitive Benefits of Two-Handed Input: An Experimental Study. *ACM Transactions on Computer-Human Interaction (TOCHI)*. 5, 4 (1999), 326–359.

[11] Malacria, S., Lecolinet, E. and Guiard, Y. 2010. Clutch-Free Panning and Integrated Pan-Zoom Control on Touch-Sensitive Surfaces: The CycloStar Approach. In Proc. *CHI'10,* ACM, 2615–2624.

[12] Mikulecky, K., Hancock, M., Brosz, J. and Carpendale, S. 2011. Exploring Physical Information Cloth on a Multitouch Table. In Proc. *ITS'11,* ACM, 140–149.

[13] Nekrasovski, D., Bodnar, A., McGrenere, J., Guimbretière, F. and Munzner, T. 2006. An Evaluation of Pan&Zoom and Rubber Sheet Navigation with and without an Overview. In Proc. *CHI'06,* ACM, 11–20.

[14] Pietriga, E., Bau, O. and Appert, C. 2010. Representation-Independent In-Place Magnification with Sigma Lenses. *IEEE Transactions on Visualization and Computer Graphics*. 16, 3 (2010), 455–467.

[15] Plaisant, C., Carr, D. and Shneiderman, B. 1995. Image-Browser Taxonomy and Guidelines for Designers. *IEEE Software*. 12, 2 (1995), 21–32.

[16] Plaisant, C., Milash, B., Rose, A., Widoff, S. and Shneiderman, B. 1996. LifeLines: Visualizing Personal Histories. In Proc. *CHI'96,* ACM, 221–227.

[17] Plumlee, M.D. and Ware, C. 2006. Zooming Versus Multiple Window Interfaces: Cognitive Costs of Visual Comparisons. *ACM Transactions on Computer-Human Interaction (TOCHI)*. 13, 2 (2006), 179–209.

[18] Sarkar, M., Snibbe, S.S., Tversky, O.J. and Reiss, S.P. 1993. Stretching the Rubber Sheet: A Metaphor for Viewing Large Layouts on Small Screens. In Proc. *UIST'93*, ACM, 81–91.

[19] Shneiderman, B. 1996. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In Proc. *VL'96*, IEEE, 336–343.

[20] Shoemaker, G. and Gutwin, C. 2007. Supporting Multi-Point Interaction in Visual Workspaces. In Proc. *CHI'07,* ACM, 999–1008.

[21] Vernier, F. and Shen, C. 2011. Multitouch Magic Fisheye: Precise interaction with dense data on tabletop. *DEXIS 2011 Workshop at ITS'11* (2011), 6–9.

[22] Voida, S., Tobiasz, M., Stromer, J., Isenberg, P. and Carpendale, S. 2009. Getting Practical with Interactive Tabletop Displays: Designing for Dense Data, "Fat Fingers," Diverse Interactions, and Face-to-Face Collaboration. In Proc. *ITS'09*, ACM, 109–116.

[23] Ware, C. and Lewis, M. 1995. The DragMag Image Magnifier. In Proc. *CHI'95,* ACM, 407–408.