T U M

I N S T I T U T F Ü R I N F O R M A T I K

Beyond Pinch-to-Zoom: Exploring Alternative Multi-touch Gestures for Map Interaction

Eva Artinger, Martin Schanzenbach, Florian Echtler,
Tayfur Coskun, Simon Nestler, Gudrun Klinker

TUM-I1006
Oktober 10

T E C H N I S C H E U N I V E R S I T Ä T M Ü N C H E N

TUM-INFO-10-I1006-0/1.-FI

Druck: Institut für Informatik der
Technischen Universität München

Beyond Pinch-to-Zoom: Exploring Alternative Multi-touch Gestures for Map Interaction

Eva Artinger *Martin Schanzenbach* *Florian Echtler* *Simon Nestler* *Tayfur Coskun* *Gudrun Klinker*

Institut für Informatik
Technische Universität München
Garching, Germany
{artingee,echtler,schanzen,nestler, coskun,klinker}@in.tum.de

ABSTRACT

Interaction with virtual maps is a common task on tabletop interfaces, particularly in the context of command-and-control applications. In nearly all cases, widely known gestures such as pinch-to-zoom are employed. To explore alternatives and variations of this mode of interaction, we have defined five alternative gesture sets for the tasks of modifying the map view and selecting map objects in an emergency management scenario. We present the results of an explorative study conducted with user interface experts, domain experts and inexperienced randomly selected users.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Design, Human Factors, Experimentation

Keywords: tabletop interface, gestures, map interaction.

INTRODUCTION

In the last few years, interactive surfaces have steadily been gaining attention in research and industry. Scenarios which use these interface have so far focused on entertainment, infotainment and visualization, more serious applications are beginning to emerge. An area where tabletop interfaces in particular are showing promise is that of command-and-control, usually within one of the two larger contexts of military operations or emergency management.

In this paper, we will look at the latter scenario where several high-ranking emergency responders are tasked with planning and coordinating a suitable reaction to a mass casualty incident (MCI). This task has several requirements which create a highly demanding environment for any potential user interface: the participants are under severe time pressure and considerable stress, yet have to collaborate effectively to ensure the best possible outcome of the situation. In most cases, this collaboration takes place in a command center around a central map of the affected area, thereby offering a natural application scenario for a virtual map on a tabletop interface.

Figure 1: Map interaction with an alternative gesture set: zooming through a spiral gesture.

To place as little additional mental load as possible on the users, any interaction with this map interface should be natural and intuitive. Gestural commands, particularly those which are based on real-world metaphors, are therefore a promising mode of interaction. Interfaces which have already been presented rely mostly on a small set of gestures such as pinch-to-zoom.

However, one question remains - are these common gestures really the best way to interact with a virtual map? We have addressed this question by assembling five alternative sets of potential gestures from various sources. Each of these sets contains gestures for the four operations of panning, zooming, rotating and selecting. In a formative study, we have evaluated these sets with users from various backgrounds: user interface experts from our research group, domain experts from the university's fire department and inexperienced randomly selected students. The subjective impressions and opinions of the users offer valuable insight into potential improvements.

RELATED WORK

In many commercial and research applications of interactive surfaces, a small set of core gestures are used time and again. Most of these gestures deal with spatial manipulation in two dimensions, i.e. translation, rotation and scaling. Popular recent examples include the iPhone and iPad products from

Apple [7] or Microsoft Surface[1]. While these gestures have been popularized in 2005 by Han [5], their origins can be traced back to as early as 1985 when Krueger et al. presented a pinch-to-zoom gesture in Videoplace [8].

While this core set probably accounts for the vast majority of gestures used on interactive surfaces, some researchers have attempted to look beyond what is already presented in literature. One example is given by Epps et al. [2] in which users were instructed to perform certain actions with User Interface (UI) element mockups without being told how to do so. An important finding of this study was that there are noticeable differences between individuals' gestures preferences. Similar results were obtained by Wobbrock et al. [18] with a different method: users were shown the result (e.g. movement of a virtual object) and had to choose a suitable gesture for the result.

The topic of interaction with virtual maps on interactive surfaces is also addressed in a number of publications. Probably one of the best-known works is DTLens by Forlines et al. [3] which focuses on the concept of "lenses" that allow several people to simultaneously view details in different parts of the map. A combination of gesture and speech input is used by Tse et al. [17] to control a commercial map application and a computer game, while Schöning et al. [14] have extended map interaction by enabling the user to also execute gestures with their feet.

A related direction of research is focused on the application of novel user interfaces in the context of emergency management or, more generally, command-and-control scenarios. For example, Micire et al. have examined preferred user gestures in the context of robot control for surveillance [9]. Another work presented by Rauschert et al. [12] places particular focus on queries related to geospatial data and tries to provide easy access through a multimodal interface based on speech and gestures.

ALTERNATIVE GESTURE SETS

Design Strategies

Good gestural interfaces have very similar characteristics to any other well designed system [13]. Gestures for map interaction should fulfill the same requirements as any other good user interface. Ben Shneiderman has defined eight golden rules of user interfaces [16] and Don Norman describes several important design principles [11]. If we transfer those rules and principles for general user interfaces to gesture based user interfaces, then gestures must be simple, straightforward, easy to remember, consistent and distinguishable. Gestures should be based on knowledge in the world and in the user's head. Natural mappings and metaphors simplify the recollection how gestures and actions are connected.

Table 1 presents an overview of gestures and the applied design strategies, which are described in the following.

We used the following strategies for defining the gestures:

Gestures based on related research results: Good gestures for tabletop interaction were found already by several researchers. Some of these gestures are suitable for map interaction as well. Jeff Han presented gestures for panning, zooming and rotating. He used two fingers pinching to shrink and spreading to enlarge. Rotation is done using two finger as well, where the pivot point is between the fingers [5].

Wu et al. present different gestures for interacting with a prototype room furniture layout application, called Room-Planner. Objects are rotated with two fingers. One finger defines the center of rotation, while the second specifies the angle [19].

Other results are given by a user centered evaluation about gestures for tabletop interaction. It shows that users don't mind how many fingers are touching the table [18]. Thus, the number of fingers are an inappropriate feature for the differentiation of actions. Considering that, those gestures are ideal, which can be performed with an arbitrary number of fingers.

Metaphors and natural mappings: User interfaces are often based on the idea of metaphors, because this approach reduces the mental load. Metaphors connect the well-known with the new - they create a connection between the real and the virtual world. For example deleting a file is done by moving it into the trash-bin. However, metaphors have to be selected carefully, as cultural differences can lead to different interpretations. A natural mapping sets a proper relation between controls and movement. Metaphors and natural mappings reduce the mental load to perform a task [16].

In our application we use a spiral as a natural mapping for zooming. Going along a spiral is like zooming through space. If the spiral is traced inwards the map is zoomed in and vice versa.

Transfer of desktop concepts: We assume that all of our users have experience in using a Desktop PC with a mouse. Based on this knowledge, interaction techniques are transferred from the Desktop PC to the tabletop interface.

To select multiple items on a desktop one method is to hold the Control key and to click on one item after the other. We transfer this "hold and tap" method to the table. The first item to select has to be held with one finger, while further items can be selected by tapping one after the other with another finger on the other hand.

Another selection method for several items is to draw a rubber band rectangle around the items. One edge of the rubber band rectangle sticks to the mouse cursor while the user holds the mouse button, making it possible to adjust the rectangle's dimension. This method is a common technique to select files for example in a file explorer or on the desktop. On the table objects are selected by drawing a semi-transparent rectangle around the objects.

Transfer of software concepts: Other popular interaction techniques are derived not from the desktop metaphor, but from well-known Software and Web-Applications like Google Maps and Adobe Photoshop. Knowledge from those applications is transferred to the tabletop interface.

[1] `http://www.microsoft.com/surface`

Operation	Individual gesture	Research	Metaphor	Desktop	Software
Pan	Sticky finger		x		
	Flick		x		
Zoom	Pinch	x			
	Spiral		x		
	Window and double tap				x
Rotate	Circle	x			
	Around center rotation	x			
	Pivot point rotation	x			
Select	Lasso				x
	Hold & Tap			x	
	Rectangle			x	

Table 1: Overview of the individual gestures and their origin in the design strategies.

We adopted the Google slider for zooming and integrated it in a control widget, which allows to pan, zoom and rotate the map. Adobe Photoshop has a zoom function where a rubber band rectangle specifies the area, which has to be enlarged. Everything inside the defined rectangle is scaled up to fit the size of the Photoshop window. Zooming out is realized by pressing the Control key in the zoom mode and clicking to the image. On every click the image is then stepwise zoomed out.

On the multi-touch table the area which should be enlarged is defined similar to the zoom-in in Photoshop. As there is no Control key on the tabletop zooming-out is done by double clicking, or rather double tapping.

Individual gestures

We created a pool of individual gestures for all four operations of panning, zooming, rotating and selecting.

Gestures for panning Moving the map is probably the most frequently used feature of a map application. Consequently, the gesture, which triggers the translation, should be simple and easy to remember.

Sticky finger: One or more fingers move the map. The finger sticks to the map and drags it along when it is moved.

Flick: The *Flick* gesture is a slight modification of the *Sticky Finger*. When the finger is removed from the surface the map continues sliding in the direction it was moved. It might become more difficult to accurately position the map. But on the other hand moving from one side of the map to another is much easier and faster than with *Sticky finger*, because the map moves automatically in the desired direction.

Gestures for zooming Zooming is one of the big advantages of a digital map over a paper map, which is not scalable. Zooming is, after the translation, a task which is done very frequently.

Pinch: The pinch-to-zoom is often found on recent multi-touch devices, like the iPhone or some Android devices. This gesture can be performed with two fingers using either one hand or two. One or two hand manipulation is a matter of personal preference and size of hardware. The action for zooming-out consists of two fingers coming closer to each other. Zooming-in is done by the reverse pinch gesture: two fingers spreading.

Free pinch: This gesture is a modification of the pinch gesture. *Free pinch* is executed using an arbitrary number of fingers (2-5 per hand), which breaks the limitation of the fingers of the original pinch gesture.

Spiral: The *Spiral* is triggered by moving one finger on the surface. Moving inwards along the spiral zooms in, moving outwards zooms out. The spiral can be seen in figure 1

Window and double tap: To zoom-in the user draws a rubber band rectangle onto the screen. The frame is drawn like the translucent rubber band rectangle used in desktops and file managers to select items. As soon as the user has selected the area of interest the map is zoomed so that the area fills the screen. This can be done so often until the maximum zoom level of the map is reached. Zooming out is stepwise possible. The user can double tap onto the screen to revert the last scale operation. A history of scale operations is recorded so that successive double-taps can be handled.

Contrary to the aforementioned gestures continuous scaling is not possible, making small adjustments to the scale factor infeasible.

Gestures for rotating Rotating a map is especially important on a multi-touch table, when not only one person is using the map. People standing around the map might want to see it correctly aligned.

Around center rotation: This gesture is very similar to the already mentioned pinch-to-zoom gesture and was also shown by Jeff Han. Two fingers are moved clockwise or counterclockwise while the map changes the orientation. During the movement the map rotates around the center of rotation, which lies between the fingers.

Pivot point rotation: This gesture is slightly different to the first one. The center of rotation is not between the fingers, but it is specified by the position of the first finger, which touches the tabletop. According to Guiard [4] the non-dominant hand sets the reference frame and is the first which starts the action. In our case it sets the pivot point for the rotation and holds it, while the dominant hand defines the degree of rotation.

Circle rotation: The last rotation gesture presented here is based on a one finger circling motion. As soon as one finger starts dragging the map is rotated around a fixed point close to the finger. For as long as the gesture is in progress the center of rotation stays fixed.

Gestures for selecting Selecting and highlighting items is another core feature, which is implemented in our application.

Rectangle: A rubber band rectangle, similar to the one described in the *Window and double tap* gesture for zooming, can be used for selection as well. In the selection mode every object inside of the rectangle is selected. This method is derived from the rubber band selection on a desktop or file manager. We think that it is an easy way to perform and to remember the gesture. However, one of the main drawbacks of this function is, that it is not possible to draw a rectangle around the desired objects. An unwanted object may be positioned in-between and might be selected accidentally.

Lasso: This type of selection is common in graphics programs like Adobe Photoshop. In contrast to the rubber band, this method allows the user to be more accurate with the selection area. Though there are two disadvantages, when using the *Lasso* selection. At the beginning it might feel a little slower and more difficult, because it takes longer to select the same amount of items compared to the *Rectangle*. So this describes a trade off between accurate area selection with the *Lasso* and the more comfortable, faster selection with a rubber band rectangle.

Hold and tap: We transferred another desktop interaction technique for selection. A common way to select items on the desktop is to hold the Control key while clicking on one item after the other. This "hold and tap" method is performed in a similar manner on the table. The first item is selected with one hand and then held while further items can be selected by tapping one after the other with the second hand. *Hold and tap* is a good method to select a couple of items, but if the number of items increases, the time to execute the selection rises as well.

Gesture Sets

We combined the aforementioned gestures to five different gesture sets for evaluation and testing purposes. However, some gestures can not be combined with other gestures. The interpretation of the gesture takes place during the execution. Due to that fact, the system has to know right from the beginning what gesture is performed to respond correctly. We have to consider that only those gestures are put in one set, which can be detected immediately through unique characteristics, like the numbers of fingers or the position of the fingers to each other. Gesture sets are shown in figure 2.

Default: This set consists of the *Sticky finger*, *Pinch* and *Around center rotation* gestures. They are the most widely used gestures in recent multi-touch devices. The *Rectangle* method is chosen for selection. It is the only gesture, in this set, which is activated with 3 fingers and does not get in conflict with other gestures.

Circle: A single finger is used to rotate the map. A circling motion triggers the gesture and the rotation of the map occurs accordingly. The center of rotation is close to the finger performing the gesture. Moving the map is possible by dragging the map using two fingers. To zoom in or out of the map the pinch-to-zoom gesture is available. Selection is done with the *Lasso*, which can be drawn using three fingers.

Spiral: This set is named after the *Spiral zoom* gesture. Other gestures included are the translation with two fingers, *Between the finger rotation* and *Rectangle* for selection. The translation and rotation gestures are both performed by using two fingers, but it is possible to distinguish between these two by taking the proximity of the fingers into account. If the amount of space between the fingers is within a certain threshold the move gesture is triggered. If the gap is bigger the rotation gesture is triggered.

Fly: This set is characterized by the zooming method *Window and tap*. The effect of *Window and tap* is like flying into the window. Hence, we call the set *Fly*. Translation is triggered by placing two fingers in close proximity to one another. Rotation is done using the *Pivot point rotation*. The *Lasso* gesture for item selection is also found in this set. Since it is triggered and executed using three fingers, it does not get in conflict with the other gestures.

Free: Wobbrock et al. [18] found that most users do not pay attention to the number of fingers touching the surface. In response to the problem, we have implemented the gesture set *Free*, which is based on the *Default* set. In contrary to the latter all gestures of the *Free* set can be performed with one up to ten fingers. The user is not restricted to any fixed numbers. Removing this artificial boundary might help the user to avoid errors. *Lasso* and *Rectangle* are no longer possible to execute because the gestures would be indistinguishable for the system. So the only remaining method of selection is *Hold and Tap*.

Custom: We also implemented a custom set, where the user can combine the personal preferred set of gestures. As we mentioned at the beginning of this section, it is not possible to combine some gestures. If the user starts to select the first gesture all other gestures which are not compatible are no longer selectable.

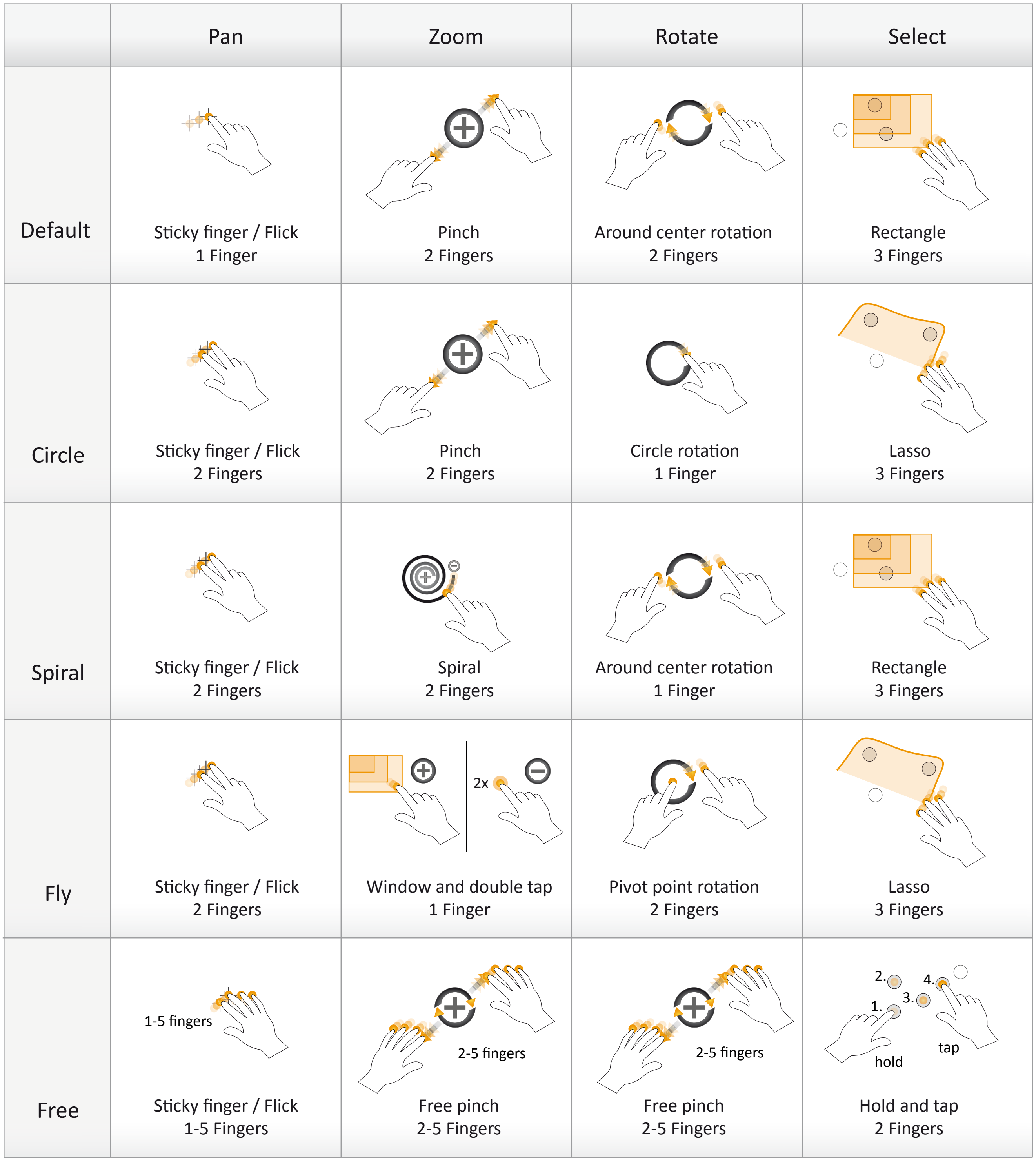


Figure 2: Five different gesture sets with support for translation, scaling, rotation and selection. The icons are also used as help graphics. The visual feedback associated with every gesture is also illustrated.

Control widget

The control widget or button widget consists of several buttons combined to a pointed oval as shown in figure 3. The widget merges translation, rotation and zooming functionality in one shape. Eight directions are possible for translation. Two buttons can be pressed for rotation, one for clockwise rotation on the right side and another one for counterclockwise rotation on the left side. If the buttons are positioned in this way the rotation feels more natural. We came to that conclusion, when we used the widget first time.

The slider is similar to a Google Maps slider. It can be used for continuous zooming. Stepwise zooming is possible with two buttons above and below the slider. Those buttons also indicate what effect the movement of the slider in this direction will have. For instance moving the slider towards the button with the "+" will scale the map up and vice versa.

By default the widget is not visible on the map. Tapping with four fingers makes it appear below the user's fingers. Tapping on the map hides the widget. Items can be selected with *Hold and tap*.

Visual Feedback

Our application gives immediate visual feedback during the interaction. The instant response of the system provides the user with an indication of the system state. It makes the outcome of actions apparent. When designing the feedback icons it has to be considered that the feedback visualization should be recognizable regardless of the user's point of view. As a result, symmetric and simple graphics are designed to fulfill this requirement. The feedback graphics can be seen in figure 2.

Touchblobs Feedback is given in form of orange semitransparent circles when touching the surface of the table. This information makes the user aware of the number of detected touches. The user can notice if the system is responding correctly or not.

Translate: A cross sticks to the finger when the map is translated.

Zoom: The symbol for zooming is often a magnifying glass. We reduced it to a circle with a plus or minus sign inside. This design ensures that the graphical appearance is consistent from every side of the table. Arrows indicate the direction of the finger movement.

Rotate: We have designed three slightly different rotation illustrations. All of them consist of a circle and at least one arrow showing the direction of the rotation. A circle with two arrows is used for the *Around center rotation* with the center of rotation in between and only one arrow for the *Circle rotation*. An orange point marks the pivot point in the *Pivot point rotation* surrounded by a circle with one arrow.

Select: The selected area of the *Rectangle* method is visualized with a semitransparent rectangle. A polygon is drawn with the *Lasso* method. A semitransparent overlay is shown between the end and the start point. Selected objects are highlighted.

Figure 4: Feedback during *Pivot point rotation*.

Figure 5: Help system for the *Fly* set. Illustrations indicate how to perform each gesture for translating, zooming, selecting and rotating the map (from left to right.)

Help

Don Norman stated that a user interface has to bridge the gulf of execution and evaluation. What he means by the gulf of execution is the difference between the intentions of the users and what the system allows them to do or how well the system supports those actions [11]. We developed a help functionality, which visualizes available actions, so people are aware of possible actions and their execution.

Furthermore one of Shneiderman's eight golden rules of interface design is to *"Reduce short term memory load"* [16]. The help system reduces the memory load as well as it bridges the gulf mentioned before. The user can always rely on the help system if the recall of the gestures is not correctly or not possible at all. The help can be called with a five finger tap on the table. Tapping again hides it.

All available gestures are illustrated with icons depicting hands that perform the gesture including the resulting visual feedback. The help for the *Fly* set can be seen in figure 5. Apart from assisting the user, the help functionality has advantages for the developer as well. The frequency and the duration of the user's need for help can be tracked. Therefore, the call of the help gives an indication about the memorability of the gestures.

FORMATIVE USER STUDY

In order to compare the gesture sets, we conducted a user study to find answers to the following questions:

- Is the application self-explanatory and intuitive?
- How difficult are the gestures to learn, to perform and to remember?
- Are the icons of the help menu easy to understand?

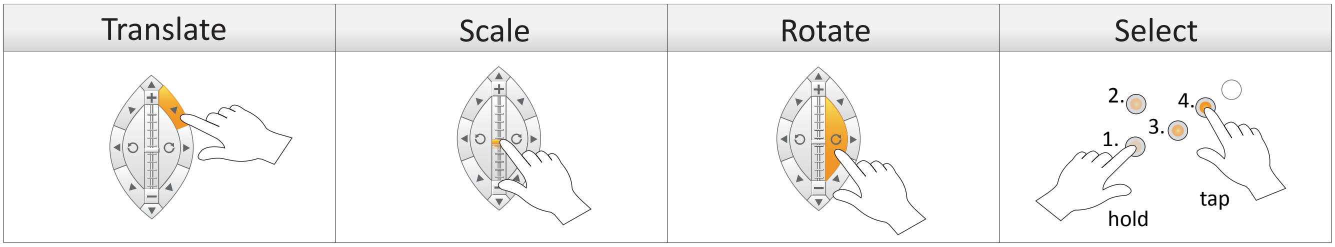


Figure 3: The control widget combines translation, rotation and zooming functionality in one shape.

- Are the gestures suitable for left as well as right-handed people?
- What is the best suited interaction technique?

Participants

Nielson has shown that a small sampling size is enough to identify the most critical usability problems in an early stage [10]. Additionally Schwerdtfeger states that a small and diverse test group consisting of user interface experts, domain experts and inexperienced random users enhances the problem finding as well [15]. Hence, we selected six participants for the study: Two user interface experts, two inexperienced students and two domain experts. The two domain experts are from the fire department of the Technische Universitaet Muenchen (Feuerwehr TUM).

The subjects' age ranges from 21 to 49, one female and five male, two left-handed and four right-handed. All Participants are familiar with Desktop PCs. Four participants have used an iPhone or an iPod Touch, one of them used an Android based smartphone. Two users have experience with multi-touch tables. Another two participants are using touch interfaces daily, like phones or touch displays, the other four once a month or less. Five participants know Google Maps and two are familiar with Open Street Map.

Procedure

Participants had to execute realistic tasks, which are very common and are often performed with a map. These nine tasks included panning, scaling and rotating the map. Another task was also the selection of single and multiple items, in our case patients. The patients were illustrated as pins. The tasks were all written down on a sheet of paper which could be read during the whole evaluation.

At the beginning all participants had to fill in a questionnaire concerning demographic data and their experience with touch. Afterwards each participant had to complete all nine tasks in a fixed order using each of the five gesture sets, one after the other. That means by the end every user had done all nine tasks five times. We used a within-subject design and changed the order of the gesture sets randomly for each user. The system was not introduced to the test persons, because one of our goals was to see how self-explanatory and intuitive the application is. Participants were asked to think aloud during the evaluation.

After solving the nine tasks with one gesture set a questionnaire had to be filled out. Three questions were asked concerning the ease of learning, ease of performing as well as the ease of remembering of each set, using a five point-likert scale. Furthermore users could give comments on their preferred gestures of each set. Afterwards they had to fill in a standard System Usability Scale (SUS) questionnaire [1].

Each session lasted between one and two hours. At the end of each session the participant could select the preferred gestures for each operation. Furthermore the test person could give a final rating of the application by filling in an AttrakDiff [6] questionnaire. The latter allows us to assess the hedonic and pragmatic quality of the application. Video and audio recordings were made in all sessions.

Apparatus

The hardware on which our study was conducted is a rear-projected FTIR-based [5] multitouch table with a screen diagonal of about 1.30 m. As the interaction surface is situated at a height of approximately 0.9 m, the system can be comfortably operated by one or more persons standing beside the table. The image is projected at a resolution of 1024x768 pixels while the infrared camera used for touch detection has a sensor size of 720x576 pixels. These dimensions result in a sensor resolution of approximately 15 DPI and a display resolution of approximately 25 DPI. Although these values are not particularly high, they are nevertheless sufficient for operation by a person standing at the table.

RESULTS

This section described our experimental results and gives a detailed discussion of the findings. The presented results concern both our predefined gesture sets and individual preferences of single gestures for the operations moving, scaling, rotating and selecting.

Predefined gesture set preference

We analyzed the SUS questionnaires as well as questions regarding ease of learning, ease of performing and ease of remembering in order to find the most preferred gesture set from our gesture set pool.

Favored sets Free and Default All five gesture sets and the widget have been evaluated using a SUS questionnaire. In general the calculated SUS values range from 0 - 100, where 100 is the optimum. The questionnaire assesses the usability of the system. In boxplot 6 the results of the SUS questionnaire are shown. The highest rated gesture sets are *Default*, *Free* and *Spiral* respectively. The more ”traditional” gestures achieved the highest overall scores. Surprisingly the rather exotic gesture set *Spiral* is also rated well above *Fly* and *Circle*. The scores for the *Free* set are scattered from very low to

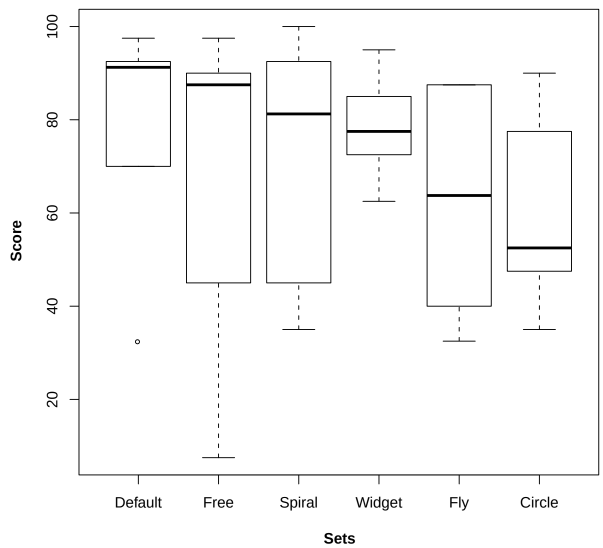


Figure 6: SUS Score for each gesture set.

very high values. A reason might be that during some of the tests multiple unwanted gestures had been activated due to the arbitrary number of fingers triggering each action. Therefore, the system could not detect the correct gesture clearly.

The values for the *Spiral* set are more consistent and the median values of the *Spiral* and *Free* are not that far away from each other. The widget set takes fourth place, however, it does not represent gestures but rather an alternative to gestures. The *Fly* and *Circle* sets are the most unfavored sets. Both sets have the same selection gesture namely *Lasso* in common. During the tests we observed that people had particularly problems with using the *Lasso*. The *Lasso* gesture had to be performed with three fingers, but our system did not always detect all three fingers. Therefore, the selection was hard to perform, which could lead to the low rating of those two sets.

Ease of learning and ease of performing The results of our custom questions on ease of learning and performing support to a certain degree the SUS scores. In figure 7 the ease of learning and performing the gestures is presented. The results show that for both the *Default* and the *Free* sets were very easy to learn. In case of the *Default* set the execution of the gestures was also rated very easy, except for the selection gesture, which is the *Rectangle*. On the other hand, the *Hold and Tap* selection gesture in the Free set seems to be extremely easy both to learn and to use. In both questionnaires, SUS and custom, the *Default* set and the *Free* set are rated best.

However, the results from the *Spiral* set are not consistent with the SUS result. Three actions namely move, rotate and select were very easy to learn, but the *Rectangle* was not that easy to use. The problem with the three finger detection comes again into consideration. The spiral gesture for zooming was not rated very well in terms of performing and learning. The Fly set obviously is evaluated as bad as in the SUS. Executing and learning the zoom and selection gestures apparently was very difficult. The other two gestures might have been easier to learn but still hard to execute. Surprisingly, most of the gestures in the *Circle* set were indeed quite easy to learn and use. This is not at all consistent with its SUS score considering that it lies even below the score of the Fly set.

Ease of remembering *Default* and *Free* are easier to remember than *Fly*, *Spiral* and *Circle*. The latter three are all rated the same.

We tracked users who called the help function for different gestures. The results show that the *Spiral* and *Fly* sets had been called eight and nine times respectively, for all user tests. This shows that the *Spiral* and *Fly* sets require most support to use them. Therefore, this confirms and supports our finding from the results presented above.

Control widget or gestures

Users were asked to state whether they liked to use the gestures or the control widget as interaction method. Three participants voted for gestures only. Two participants wanted to have a combination of widget and gestures, while one user's preference was to use only the widget. However, the widget could be provided as an additional interaction method to the gestures, if the user prefers the widget.

Individual gestures and users' preferences

We asked users to vote for their favorite gestures after working with all sets. The following describes results the results:

Panning For moving the map only three gestures were available. They only differed in the number of fingers that are used to perform the gesture. In the end the sticky finger as found in the *Default* set and the move gesture found in the *Free* set were the clear favorites.

Zooming By taking a look at the votes for scaling the pinch-to-zoom gestures are in front but only by one vote each. Consequently no clear favorites emerged in this case.

Rotating To our surprise the *Circle rotation* gesture received a number of votes. It is together with the *Around center rotation* the best gesture for rotation.

Selecting The *Lasso* and *Rectangle* gestures got the same amount of votes. However, *Hold and tap* is clearly breaking away with a big difference, leaving the other two gestures behind by a large margin.

Hedonic and pragmatic quality

The AttrakDiff questionnaire provides some insight on the attractiveness of an interactive system. It measures the attractiveness, the hedonic, and the pragmatic quality. Hedonic quality (HQ) is based on human needs for stimulation (HQ-S) and identification (HQ-I). The pragmatic quality (PQ) is a grade for how successfully the user achieves his goals using the product [6].

In our study we evaluated the whole application with the AttrakDiff and not each gesture set individually. Figure 8 (a) shows the four aspects in a graph. The pragmatic quality is below HQ-I, HQ-S and attractiveness (ATT). This leads to the conclusion that the application was in general attractive

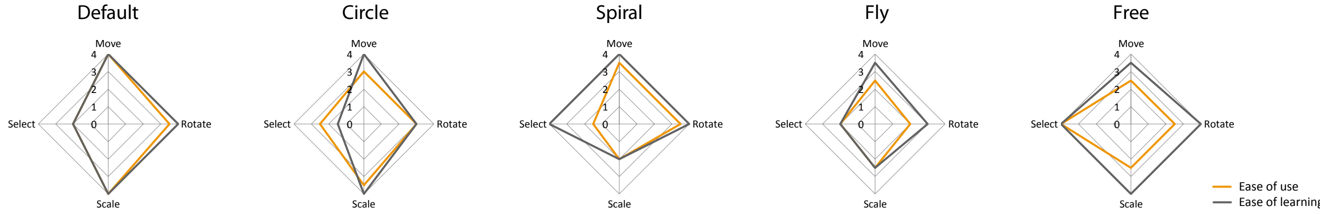
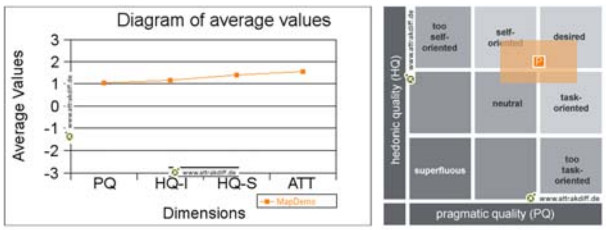


Figure 7: Net Graphs showing how easy the gestures were to learn and to perform. The higher the value the better.



(a) (b)

Figure 8: a) Mean value of attractiveness (ATT), hedonic quality stimulation (HQ-S) and identification (HQ-I), b) Portfolio with average values of the dimensions PQ and HQ and the confidence rectangle of the product

to the user. However, the pragmatic quality has room for improvement. The focus of this application is mainly on the usability and with it pragmatic quality. Therefore, this should be improved.

Figure 8 (b) shows the average values of PQ and HQ. The Medium value of the dimensions (P) is in the upper left corner and as ”rather desired” rated according to the AttrakDiff evaluation report. The size of the confidence rectangle shows the consensus of the opinion. In our case the dimensions are rather large due to our limited sample size.

DISCUSSION & FUTURE WORK

User feedback

Users had the option to comment freely on the application and the gestures. From our observation, we noticed that users accidentally activated gestures that they did not intend to perform, and in some cases users interrupted their experimental tasks with and accidental gesture. For example, in the process of selecting items using any of the three finger gestures it frequently happened that users accidentally rotated or moved the map, because not all 3 fingers were detected by the system.

The majority users reported that using three fingers for *Lasso* or rubber band rectangle selection is impractical. The *Hold and tap* gesture received a lot more positive comments. Users suggested that the selection gestures should be activated using three fingers and continued using only a single finger. However, this was actually already possible yet unknown to the user because the help icon did not explicitly show this feature.

The last thing to note is that the widget received very mixed comments. It was characterized as being ”very easy to use” and ”very clear” but it was also reported that it is ”slow to use”. A user interface expert added that the widget removes the aspect of direct interaction from the interface, because interaction is no longer done with the map but with the widget. This forces the user to constantly change his focus between the tool and the map.

Recommendations for gestural interaction

Based on our quantitative and qualitative data as well as our observations during the evaluation we have some recommendations for future gestural interfaces.

Don’t differentiate by the number of fingers As shown also by [18] the number of fingers is not the best choice to distinguish between triggered actions. Gestures which could be performed with any number of fingers like *Free* are best.

Provide help functionality To reduce memory load well-known gestures should be used. A help system, which can be easily accessed and understood in a second, helps the user, if gestures have been forgotten.

Continue triggered actions Once an action is triggered, it should not be switched during the execution of the gesture, even if the system is no longer detecting the same number of fingers. Users tend to take their finger off when starting a new interaction. Thus, new actions can be started as soon as the user removes all fingers and touches the surface again.

Offer different gestures for the same action The system should offer multiple alternative gestures for one action. For example there could be different gestures for selecting a huge group of objects or only some of them. *Lasso* is a good tool to select a lot of items whereas *Hold and Tap* is better for a few items. The same applies to zooming. Depending on the required accuracy of zooming different methods are best.

Give fallback options Our application should be used in very critical situations - during mass casualty incidents. Hence we need a system with high reliability. If for some reasons the gestural interaction does not work as expected, fallback options are needed. The widget could be one of these options, other inputs could be a digital pen or a conventional mouse.

Provide shortcuts Shortcuts allow the user to do specific actions faster. A very valuable shortcut are buttons, which allow to jump to a certain position. In our application we placed arrows as hints for where the next patient, which can’t be seen currently, is located on the map. Those hints are positioned at the edges of the screen. Some users tried to tap them in the hope that the application moves to that patient automatically.

Another example where shortcuts would be useful is the *Spiral* gesture. The *Spiral* does not give the possibility to "jump" to a certain zoom level by tapping somewhere onto the *Spiral*. Shortcuts are a useful feature, which saves time.

Future work

Future work includes the development of single-user application with an optimized gesture set. We think an improved *Free* set is the best choice. Moving, scaling and rotating the map should be possible with an arbitrary number of fingers. The effect that suddenly unwanted gestures are activated should be avoided as well. A proper mechanism that only allows one gesture to be active at the same time until all fingers are removed from the multi-touch surface is needed. For the selection we chose the *Hold and tap* gesture extended by the *Lasso* as an additional gesture for selecting many objects.

As soon as the single-user application is robust we will extend our system to a multi-user application. Several incident commanders should be able to work simultaneously at the tabletop surface in future.

The interaction techniques presented are also relevant to a broader range of tabletop applications. Moving, zooming, rotating, and selecting are very common operations and our gestures are so generic, that they are appropriate for different purposes.

CONCLUSION

Our work has explored a variety of gesture sets, which go beyond the widely used pinch-to-zoom gestures, for the interaction with a map application. The application is designed to be used in emergency situations. It gives incident commanders an overview in a mass casualty incident (MCI). An MCI is a very time-critical situation. Therefore, the user interface needs good usability in order to support and not to distract incident commanders. In order to find the most appropriate gestures for our target group, we designed five different gesture sets and a button widget. Each of these sets contains gestures for the four operations of panning, rotating, zooming and selecting. "Traditional" gestures like pinch-to-zoom are compared to new and promising gestures in a formative user study. Based on our results we give recommendations for further gestural applications. This work represents a first step in bringing interactive surfaces closer to a new target audience, which has to solve serious problems.

ACKNOWLEDGMENTS

We thank Feuerwehr TUM, in particular Thomas Schmidt and Axel Grasser as well as our test participants.

REFERENCES

1. J. Brooke. SUS: A quick and dirty usability scale. In P. W. Jordan, B. Weerdmeester, A. Thomas, and I. L. Mclelland, editors, *Usability evaluation in industry*. Taylor and Francis, London, 1996.

2. J. Epps, S. Lichman, and M. Wu. A study of hand shape use in tabletop gesture interaction. In *Ext. abstracts of CHI '06*, pages 748–753, 2006.

3. C. Forlines and C. Shen. DTLens: multi-user tabletop spatial data exploration. In *Proc. of UIST '05*, pages 119–122, New York, NY, USA, 2005. ACM.

4. Y. Guiard. Asymmetric division of labor in human skilled bimanual action: the kinematic chain as a model. *Journal of motor behavior*, 19(4):486–517, December 1987.

5. J. Y. Han. Low-cost multi-touch sensing through frustrated total internal reflection. In *Proc. of UIST '05*, pages 115–118, 2005.

6. M. Hassenzahl, M. Burmester, and F. Koller. AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität. In *Mensch & Computer 2003: Interaktion in Bewegung*, pages 187–196, Stuttgart, Germany, 2003. B. G. Teubner.

7. S. Jobs et al. Touch screen device, method, and graphical user interface for determining commands by applying heuristics. U.S. Patent 7,479,949, 2008.

8. M. W. Krueger, T. Gionfriddo, and K. Hinrichsen. VIDEOPLACE - an artificial reality. In *Proc. of CHI '85*, pages 35–40, 1985.

9. M. Micire, M. Desai, A. Courtemanche, K. M. Tsui, and H. A. Yanco. Analysis of natural gestures for controlling robot teams on multi-touch tabletop surfaces. In *Proc. of ITS '09*, pages 41–48, New York, NY, USA, 2009. ACM.

10. J. Nielsen and T. K. Landauer. A mathematical model of the finding of usability problems. In *CHI '93: Proc. of the INTERACT '93 and CHI '93*, pages 206–213, New York, NY, USA, 1993. ACM.

11. D. A. Norman. *The Design of Everyday Things*. Pearson Addison Wesley, Boston, MA, 2002.

12. I. Rauschert, P. Agrawal, R. Sharma, S. Fuhrmann, I. Brewer, and A. MacEachren. Designing a human-centered, multimodal gis interface to support emergency management. In *Proc. of GIS '02*, pages 119–124, New York, NY, USA, 2002. ACM.

13. D. Saffer. *Designing Gestural Interfaces: Touchscreens and Interactive Devices*. O'Reilly Media, Inc, 2008.

14. J. Schöning, F. Daiber, A. Krüger, and M. Rohs. Using hands and feet to navigate and manipulate spatial data. In *Ext. Abstracts of CHI '09*, pages 4663–4668, 2009.

15. B. Schwerdtfeger. *Pick-by-Vision:Bringing HMD-based Augmented Reality into the Warehouse*. PhD thesis, Technische Universität München, Fakultät für Informatik, München, to appear in 2010.

16. B. Shneiderman and C. Plaisant. *Designing the User Interface: Strategies for Effective Human-Computer Interaction (5th Edition)*. Pearson Addison Wesley, Boston, MA, 2010.

17. E. Tse, C. Shen, S. Greenberg, and C. Forlines. Enabling interaction with single user applications through speech and gestures on a multi-user tabletop. In *Proc. of AVI '06*, pages 336–343, New York, NY, USA, 2006. ACM.

18. J. O. Wobbrock, M. R. Morris, and A. D. Wilson. User-defined gestures for surface computing. In *Proc. of CHI '09*, pages 1083–1092, New York, NY, USA, 2009. ACM.

19. M. Wu and R. Balakrishnan. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Proc. of UIST '03*, pages 193–202, New York, NY, USA, 2003. ACM.