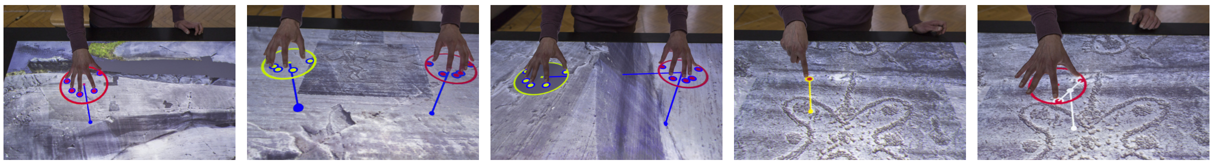
RST 3D: A Comprehensive Gesture Set for Multitouch 3D Navigation

Alexander Kulik* André Kunert* Magdalena Keil* Bernd Froehlich*

Virtual Reality and Visualization Research Group at Bauhaus-Universität Weimar



(a) Panning + 2D Rotation (b) 2D RST (c) Vertical Translation (d) Single-Finger Pointing (e) 3D Rotation

Figure 1: Multitouch 3D navigation on a large rock panel with small figures. Input gestures with one or two hands (> 2 fingers) correspond to the established 2D RST mapping (rotation, scaling, and translation) (a&b). Bimanual pinch gestures operate scaling if applied symmetrically (b) or vertical translation, if applied asymmetrically (c). Single finger input does not affect the scene, but it controls a virtual ray to point at features below the screen (d). One-handed input with two spread fingers operates full 3D rotation (e).

Abstract

We present a comprehensive multitouch input mapping for 3D navigation of multiscale 3D models. In contrast to prior work, our technique offers explicit control over 3D rotation, 3D translation, and uniform scaling with manipulative gestures that do not require graphical widgets. Our proposed technique is consistent with the established RST mapping (rotation, scaling, translation) for 2D multitouch input and follows suggestions from prior work on multitouch 3D interaction. Our implementation includes a rendering technique that can reduce perceptual conflicts of 3D touch input on stereoscopic displays. We also report on two user studies that informed the suggested interaction design and confirmed its usability.

Index Terms: Human-centered computing—Human computer interaction—Interaction techniques—Gestural input

1 Introduction

The suitability of tabletop displays for collocated collaboration has often been demonstrated (e.g. [23, 40]), especially in combination with direct multitouch input [20]. Similar setups for the collaborative exploration of genuinely three-dimensional data were enabled by multi-user 3D projection technology [1, 10] or head-based projection on a retro-reflective screen [22].

Most prior research on multitouch 3D interaction was focused on monoscopic displays that have different requirements than immersive ones with head-tracked stereo viewing. For example, scaling and distance adjustments are generally subsumed as zooming in monoscopic environments. Earlier proposed multitouch 3D interaction techniques for immersive 3D displays (e.g. [10, 15, 45]), on the other hand, did not offer direct gesture-based control for the full set of 3D transformations and uniform scaling.

We developed a comprehensive set of multitouch input gestures for multiscale 3D navigation on an immersive 3D tabletop display. Our iterative design process was guided by a large body of related work and a user study on implicit motor behavior. The resulting gesture design follows the popular approach of isomorphous input mappings that imply combined rotation, scaling, and translation input (RST) in the context of 2D user interfaces [36]. Additional degrees of freedom (DOF) are realized through mode switching based on hand postures and bimanual motor symmetry. We reduce perceptual conflicts by view-dependent clipping when the users' hands intersect virtual objects. Finally, a formal usability study was performed to validate the resulting interaction design.

Our specific motivation for this work stems from a project that required the interactive visualization and navigation of large multiscale landscapes. A 3D tabletop display appeared to be an ideal solution since one can easily start from an overview and navigate to the sites of interest. However, this appeared to be difficult for many people when using a 3D motion controller. We thus explored suitable multitouch input mappings. Our main contributions are:

- the design and implementation of the first gesture-only multitouch 3D input solution with explicit control of 3D rotation, 3D translation, and uniform scaling,
- the results of two user studies that informed and validated the suggested interaction design, and
- the development and evaluation of a cutaway rendering technique for hands that intersect with virtual objects appearing in front of the touchscreen in order to reduce perceptual conflicts.

We implemented our techniques on a 3D tabletop display with head-tracked stereo viewing for up to three users. Based on diffuse infrared (IR) illumination and the maximally stable extremal regions algorithm, the device features robust touch tracking of multiple hands and their associated fingers [12].

2 Related Work

Touch sensors capture motion input in two dimensions only and thus appear inappropriate for effective 3D interaction. Studies showed that they may nevertheless enable comparable or even better performance in 3D translation tasks than 3D motion tracking interfaces [2, 15]. More specifically, touch-input on a surface seems to enable more accuracy than direct 3D target acquisition in mid air.

Three general approaches can be distinguished: 1) widgets that offer areas or handles for different geometrical transformations [3, 7, 16, 21, 50], 2) direct touch manipulation in screen space (e.g. [11, 15, 30, 38, 39]), and 3) combinations of the latter with metaphorical manipulation gestures (e.g. [2, 17–19, 26, 31]). Please refer to [24] for a comprehensive survey of multi-touch 3D interaction techniques.

2.1 Direct 3D Manipulation in Screen Space

The paradigm of direct touch manipulation implies that finger contacts with the projected image on a touchscreen remain congruent with the corresponding 3D location in the virtual scene. In the context of 2D user interfaces, this results in combined rotation, scaling,

*e-mail:<first name>.<last name>@uni-weimar.de

and translation (RST) with motion input of two fingers [36]. Reisman et al. [39] demonstrated that the same paradigm can be extended to define corresponding 3D transformations with input from three or more fingers—despite scaling. The operation of 3D transformations through their projection into screen space may appear particularly intuitive, but it can be ambiguous [3, 39].

Hancock et al. [18] suggested limiting direct manipulation in screen space to motion input from two fingers which can always be unambiguously mapped to 3D translations and rotations around the display normal. The Sticky Tools technique implements control over the two missing rotational DOF more indirectly with motion input of a third finger anywhere on the screen. Scaling can be applied through an additional widget.

2.2 Techniques with Indirect Mappings

Widgets for 3D manipulation based on 2D touch input enable improved control and accuracy [3, 7, 16, 50], but they generally introduce visual clutter and the need to switch the focus of attention. In collaborative settings, the placement of widgets can also limit the accessibility to a single user, while direct manipulation gestures can be applied anywhere on the shared screen. As a compromise, several techniques have been suggested that apply 2D motion input on the screen directly to the displayed scene, while movements in depth and 3D rotations can be controlled slightly more indirectly, often based on comprehensible visual metaphors.

The Balloon Selection technique, for example, is an asymmetric bimanual technique where touch input of one hand directly controls the x/y position of a 3D cursor above the touch surface. Additional touch input from another hand controls the height of the cursor by adjusting the distance between both hands. The method implements the metaphor of pulling a balloon on a string that is redirected by one of either hands from vertical to horizontal movement [2].

Strothoff et al. [45] suggested the Triangle Cursor as a symmetric alternative to the balloon technique and showed that both metaphors also support rotation around the depth axis. In their study, Triangle Cursor outperformed the asymmetric Balloon Selection technique in 4 DOF docking tasks. Strothoff et al. also suggested an additional trackball widget to be operated with the respective other hand to control the missing two rotational DOF.

Other techniques for indirect depth adjustments and 3D rotations have been demonstrated without the need for explicit metaphors. DS3, for example, implements 2D translation control with one finger, while two fingers operate 3D rotation. In both cases, motion input from a further finger that is not directly touching the manipulated content controls motion in depth [31].

Liu et al. [29] later showed a more elaborate gesture recognition that offers comparable functionality with only two fingers on the screen. In their implementation, two moving fingers operate 3D translation and rotation around the depth axis, which corresponds to Sticky Tools and the Triangle Cursor. If instead one of either fingers is resting, the moving one controls tilting around the projected 3D position under the resting finger.

2.3 Visual Consistency and Mode Switching

The design of multitouch 3D input techniques often aims for visual consistency between 2D motion input and the resulting geometrical transformations in a 3D scene. Moerman et al. [34] and Marchal et al. [30], for example, argued that the traces of input gestures should correspond to the visual flow of the resulting virtual motion they are causing. Following this consideration, they developed 3D navigation techniques that constrained 3D camera movements to task-related subsets that can be expressed with 2D gestures. Marchal et al. furthermore described mode-switching between these subsets based on the recognition of the first principal movement type with two-finger input (rotation, scaling, or translation).

In a similar spirit, Cohé and Hachet [8] asked users which multitouch gesture they would intuitively apply to realize predefined transformations of a 3D cube. They observed that users take geometrical features of the object into account - either to specify the coordinate system in which their transformation is meant to be applied or to pick the object at a certain location and apply motion to the touchscreen that resembles the optical motion flow of the selected feature on the screen. Unfortunately, these observations do not facilitate the distinction of touch gestures that are meant to control 3D rotation or translation. The optical flow of both 3D transformations in screen space can be identical.

Input movements along the screen plane can describe translational movements or rotations about any screen-aligned axis [43]. It is not obvious how to distinguish between both modes. One of the most apparent differences between earlier suggested multitouch 3D input techniques is their attribution of modes to different initial postures. Single-finger gestures can only operate two DOF while up to four DOF can be expressed with two fingers. Most systems support either integral 3D translation or integral 3D rotation. Transforming two rotational DOF always affects the third one too. It is thus reasonable to support integral 3 DOF rotations with two-finger gestures [41].

Few multitouch 3D input techniques support scaling. Monoscopic displays do not convey a sense of three-dimensional size, hence scaling and depth translation have similar visual effects and are subsumed under the label of zooming. Most suggested techniques for multitouch-operated 3D translations, therefore, apply some sort of pinch gesture (varying distance between two fingers) for movements in depth. For additional adjustments of scale, on-screen widgets have been suggested [14, 18] as well as automatic adjustments [10, 32]. If both transformations are meant to be explicitly controlled with touch gestures, two types of pinch gestures must be distinguished.

Among the described multitouch 3D interaction techniques, only two enable 3D transformations with seven DOF (including scaling). Marton et al. [32] suggested semi-automated navigation. This can be beneficial in specific settings like museums, but it does not support the free exploration of 3D data. Yu et al. [50] suggested mode switching based on dedicated touch areas along the display frame. A comprehensive set of gestures, however, that enables explicit and unconstrained control over 3D translation, 3D rotation, and uniform scaling without widgets, was missing.

2.4 Touch Input on Stereoscopic Displays

In case of immersive 3D displays with stereo- and motion parallax, the apparent mismatch of motion input on a surface and the resulting 3D transformations is not the only design challenge. Displayed content with negative or positive disparity appears in front or behind the display surface, hence, disconnected from the user's touch input.

Congruency between the touching fingertips and stereoscopic graphics can only be realized for one of either eyes [35]. Valkov et al. [48] showed that people tend to touch-select virtual 3D objects at a display location between the projected images for both eyes with a bias towards the projection for the dominant eye.

Bruder et al. [5] quantified the effects of the stereoscopically perceived offset between virtual objects and the touchscreen on user performance in target acquisition tasks. Their results indicate that touch-based selection can be competitive to a baseline of direct 3D pointing if the target offset to the screen in not larger than 5-10 cm. Valkov et al. [47] furthermore showed that within a small range close to the screen surface, virtual objects can be imperceptibly be shifted in depth to avoid perceptual conflicts.

A more pragmatic and generally applicable solution is indirect touch input. It can be more comfortable and offers more freedom for the mapping of 2D motion input to 3D transformations [6, 44, 49]. However, in our case of a collaborative tabletop settings indirect mappings can be detrimental to mutual awareness [20].

For similar settings, prior work suggested illustrative visualizations that link the touch location to scene geometry below or above the screen [2, 9, 10, 15, 16, 45]. We found that this approach is practical for content behind the screen surface, but it does not solve the more apparent conflict of a touching hand occluding virtual content that is geometrically closer to the user's eyes.

3 Setup and Approach

We built a multi-user 3D tabletop display on the basis on a prototypical three-chip DLP projector, providing independent stereoscopic image pairs for three users with a resolution of 1400×1050 pixel at an image size of 1.14m×0.85m. In contrast to earlier developments [10, 27], a single 360 Hz projector serves three users with active stereo views at 60 Hz per eye. During early collaborative 3D data visualizations, a Spacemouse™ was used for 3D navigation.

Without any touch input available, we observed that participants used their fingers on the screen to indicate features of the scene below the display surface. The design of our multitouch interaction techniques, therefore, aims to retain single-finger pointing as a communication-only gesture that does not transform the displayed scene. For more convenience and unambiguous pointing gestures, the content of interest was often moved slightly above the screen, where it would perceptually interfere with the users' hands touching the display surface. We also wanted to maintain this possibility and therefore searched for solutions to the perceptual conflict.

Our developments are based on a touch sensing method based on diffuse illumination and the maximally stable extremal regions algorithm, which supports the recognition and tracking of multiple hands and their associated fingers [12]. For tracking the associated fingers of each hand, we apply the iterative closest point algorithm, which is significantly more robust to maintain the geometric relations of such correlated points [51]. This combination of methods allowed us to consider hands instead of fingers as the basic input entity.

In our implementation, each recognized hand is represented as a single input entity for 2D translation and rotation. The hand center is computed as the average position of its associated fingers in the first 250 ms after touchdown. Thereafter, each finger only contributes with relative motion input. This ensures a stable hand center even if the number of fingers changes during an interaction. The number of fingers at touchdown and their constellation can be interpreted as posture that defines input modes for each hand. Additionally, the recognized hands above the screen can be considered by the system as an occluding object that may require adaptations of the displayed content to avoid the above-described perceptual conflicts.

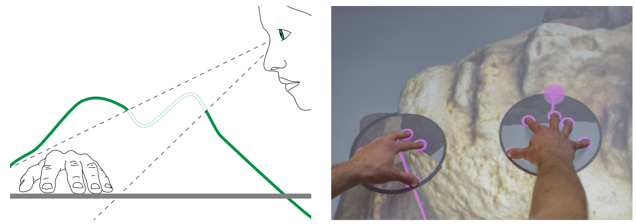
3.1 Visual Mediation of Perceptual Conflicts

Touch input at stereoscopic displays can cause perceptual conflicts due to the spatial offset between the 3D scene and the touching hand. For content behind the screen (positive disparity), the issue can be solved fairly well with virtual rays connecting the input on the surface with the 3D geometry as suggested by De la Rivière et al. [10].

If instead the 3D content appears in front of the projection screen (negative disparity), touch input causes the user to reach through the virtual geometry which results in a disturbing experience: although the virtual content is stereoscopically perceived to be in front of the hand, it becomes physically occluded. The convergence of both eyes at the hand or the surrounding geometry results in double vision of the respective other (see [5]). We suggest cutaway visualizations to alleviate this perceptual conflict (Figure 2). The geometry between a user's eye and the touching hand will not be rendered and thus the hand appears to cut a hole into the "virtually occluding" geometry.

3.2 Research Questions

Prior research on multitouch 3D input did not yet converge on a comprehensive and consistent set of gestures to realize 3D rotation, 3D translation, and uniform scaling without relying on graphical widgets. Scaling and translation orthogonal to the display are both typically operated with pinch-like gestures and thus mutually exclusive. Also the distinction of panning and tilting is ambiguous. Both are compatible with the same type of input movements [8, 30].



(a) Illustration of the technique (b) The resulting visual effect

Figure 2: Our clipping technique renders fragments of "virtually occluding" 3D surfaces (green line) invisible to maintain visual consistency.

We aimed to identify postures and quasi-postures (see [25]) that support effective mode switching between panning and tilting on the one hand and between scaling and vertical translation on the other. Additionally, we set the goal that user input under the assumption of the common 2D multitouch mapping for rotation, scaling, and translation (RST) should not affect the displayed content unexpectedly. In practice, this means to favor scaling over movements in depth (in our case vertical translations) and panning over tilting.

Some of the earlier suggested pinch gestures for translational input were implemented with asymmetric roles of two hands (e.g. [2, 31]), while most apply the same role symmetrically (e.g. [15, 18, 19, 29, 39, 45]). Scaling can also be performed symmetrically or asymmetrically. Consequently, we formulated the research question, *whether symmetric and asymmetric implementations of bimanual pinch gestures affect the motor symmetry during operation and if such implicit differences can be automatically recognized to support mode switching before or immediately after the motion onset*.

Mode switching between panning and tilting is often realized through different postures, e.g., the number of involved fingers affects the mapping of induced motion input (e.g. [11, 19, 26, 31, 44]). Unfortunately, the applied mappings are inconsistent between different implementations because the number of involved fingers has no inherent meaning related to translational or rotational movements—especially if the tracked fingers do not represent a particular posture of a single hand. Our system is capable of tracking hands with associated fingers and it was our goal to prioritize panning over tilting. This led to the research question *if a hand posture can be identified that occurs rarely during naïve operation of our tabletop touchscreen while still being comfortable and easy to perform*.

3.3 Experimental Setup

All experiments were conducted with the above described multitouch-enabled multi-user 3D tabletop. During the first study we also tested the robustness of our touch recognition and tracking methods. This led to improvements of the optical setup as well as the software. In rare cases, technical failures of our touch tracking affected user performance with the system. For performance comparisons these were filtered from the datasets. During the final usability study of our multitouch 3D navigation technique in comparison to a Spacemouse™ such tracking errors did not occur. One trial, during which the application froze, was immediately repeated.

All demonstrators and the user study were implemented using the software framework Avango-Guacamole [42]. The scene was rendered at 60 Hz and we measured about 100 ms end-to-end latency from tracking the user's motion input to visual output.

4 User Study on Bimanual Motor Symmetry

Earlier proposed pinch gestures for 3D translation can be classified into symmetric (e.g. Triangle Cursor [45] or "Sticky Fingers" [18]) and asymmetric techniques (e.g. Balloon Selection [2] or Z-Technique [31]). In the latter case, only one hand operates panning while the another one controls motion in depth by changing the distance between both. In the symmetric case, movements of both hands affect translation in all three dimensions simultaneously. We implemented both types of bimanual 3D translation gestures (*RHT* and RHT_{asym}) to study their effects on motor symmetry. For reference measures, we also included the well-known *RST* gesture from 2D user interfaces.

All three techniques built on hands instead of individual fingers as the basic input primitive. Vertical rays, emanating from the center position of each hand, were used to intersect with the scene. The rays started above the hands to enable intersections with scene content above the display surface. The resulting intersection points served as transformation contacts. A vertical line between the hand center and the contact point illustrated this relation (Figure 1(a) & 1(b)). For scaling with *RST*, the higher one of both contact points was applied as the scaling center. Its distance to the display was thus maintained during scaling. If none of both rays intersected with the scene, our technique reverted to a scaling center at the display surface.

In the asymmetric 3D translation condition (RHT_{asym}) only one hand maintained the contact with the scene which was visualized with the mentioned vertical connection line. The other hand adjusted height and rotated the scene around the connecting line. Similar to Benko and Feiner's Balloon Selection [2], this relation was illustrated with a horizontal line between both hand centers (Figure 1(c)). We distinguished the roles of both hands based on their sequence of touch. The first one operated panning, the second adjusted height.

In the case of the symmetric 3D translation technique (*RHT*), the ray intersection with the scene and the illustrating vertical connection started at the center of the connecting line between both hand centers. In our implementations we did not literally follow the visual metaphors of Balloon Selection [2] or the Triangle Cursor [45], but we focused on a consistent mapping of the applied relative motion. Increasing the distance between both hands moved the scene upwards, while a decrease moved it down—independent of whether the scene is above or below the touchscreen.

4.1 Participants, Tasks, and Dependent Variables

Eighteen students (16 - 30 years, M=23.17, SD=3.35) participated in our study on the motor symmetry of bimanual multitouch-input (8 male, 10 female). They received an allowance of 10 Euro for their participation. All participants reported using multitouch input on their smartphones every day; two reported frequent usage of a multitouch tablet device and eight participants also had some prior experience with monoscopic multitouch tabletops. Seven participants reported experience with interactive 3D graphics.

During our study the participants were asked to navigate a 3D city model on the multitouch tabletop. In this context we implemented two types of docking tasks each with 3 DOF: one required 3D translation, the other one required 2D translation and scaling. A red sphere, placed on top of a church tower in the city model had to be moved into a target position at the display center (Figure 3). The target position and tolerance was indicated with two nested wireframe spheres. The red sphere had to be placed around the inner one but inside the outer one. The difference of radii thus specified the target tolerance.

For both types of tasks, we applied two different target tolerances (7 and 14 cm) and three different lateral offsets (-30, 0, and 30 cm on the x-axis). These were combined with five vertical distances (0.3, -0.3, -0.5, -1, and -2 m) or scaling factors (1.75, 0.25, 0.167, 0.09, 0.058) respectively, which resulted in 30 different task conditions that were performed in order of increasing difficulty and repeated twice. In one distance condition the target was above the screen (0.3 m). Correspondingly one scaling condition required shrinking the model by a factor of 75%. In all other conditions, the scene had to be moved up or increased in size.

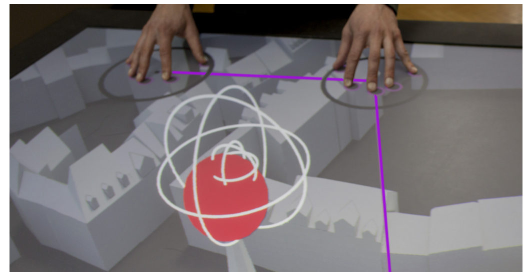


Figure 3: The 3D docking task performed with RHT_{asym}

We recorded the temporal differences between the touchdown of both hands (ΔT_{touch}) as well as the ratio of movement magnitudes shortly after initiation (*move ratio*). This data was used for the evaluation of motor symmetry with the three techniques. We also recorded the number of hands and associated fingers in every frame in order to derive rarely used hand postures that could be used for another input mode. User performance with the different techniques was captured in terms of Masliah's metric for simultaneity of control (SOC [33]) and the average time to complete a set of 3D docking tasks with controlled difficulty (TCT).

4.2 Hypotheses

We expected that the assignment of different roles to both hands in bimanual multitouch techniques implicitly affects the symmetry of motor behavior and that this could be used to distinguish between interaction modes. Considering the results of prior research comparing the Triangle Cursor (symmetric) with Balloon Selection (asymmetric) [45], lower task performance with the asymmetric technique could also be expected. More specifically, we had the following hypotheses:

H1: Symmetric bimanual input supports better performance than its asymmetric counterpart in a 3D translation task.

H2: Symmetric bimanual input fosters simultaneity of control (SOC [33]) between vertical and horizontal translation.

H3: ΔT_{touch} differs significantly between symmetric and asymmetric techniques.

H4: *move ratio* differs significantly between symmetric and asymmetric techniques.

4.3 Performance Results

Our results on task performance support H1 and H2 (Table 4.3). A t-test revealed significantly longer task completion times with RHT_{asym} compared to *RHT* (t_{17}=5.6, $p<.0001$). This may at least partly be attributed to the significantly lower simultaneity of control (SOC [33]) with the asymmetric technique (t_{17}=3.6, p=.002). The overall task completion times of 7.74 s (SD=4.51) appears relatively long for a series of 3D target acquisitions with an average index of difficulty of 4.8 (SD=1.07) [13, 46] but they compare well to results from prior research with similar interfaces [2, 15, 17, 29, 31, 50].

The performance results in the *RST* condition are not directly comparable since the task was different. Task completion times indicate, however, that the required effort was comparable. Subjective user feedback on the System Usability Scale (SUS [4]), yielded higher scores for both symmetric techniques (Table 4.3), but according to a MANOVA this difference was not significant.

4.4 Observed Motor Symmetry

During the study we recorded 6235 bimanual input gestures with *RST*, 6771 with *RHT*, and 9376 with RHT_{asym}. A MANOVA revealed significant effects of *technique* on ΔT_{touch}, the time difference between the touchdown of both hands ($F_{2,34}$=10.42, $p<0.001$, η_p^2= 0.86).

Table 1: Task completion times (TCT), simultaneity of control (SOC [33]), and system usability scores (SUS) [4] obtained with the three tested techniques.

	TCT: M (SD)	SOC: M (SD)	SUS: M (SD)
RST	7.40 s (4.26)	.29 (.22)	85.56 (10.59)
RHT	6.65 s (1.98)	.33 (.06)	85.28 (10.505)
RHT_{asym}	8.95 s (1.85)	.29 (.03)	78.61 (17.64)

Pairwise comparisons with Bonferroni-corrected alpha proved that the mean ΔT_{touch} was significantly longer with RHT_{asym} than with both symmetric techniques (both $p<0.001$).

The *move ratio* can be considered as a mode indicator after a reasonable distance of motion input has been accumulated. We evaluated our data for the three cases of bimanual pinch gestures with an accumulated distance of 7.5 mm 15 mm, and 30 mm. A MANOVA revealed significant effects of *technique* on *move ratio* ($F_{2,34}$=56.68, $p<0.001$, η_p^2= 0.77) and a significant interaction with the travelled distance ($F_{2,34}$=62.85, $p<0.001$, η_p^2= 0.79). Pairwise comparisons with Bonferroni-corrected alpha proved significant differences between RHT_{asym} and both symmetric techniques (both $p<0.001$). The obtained interaction effect indicates that the difference between symmetric and asymmetric techniques increases with traveled distance.

These results support our hypotheses H3 and H4. However, this is not a proof that any of either suggested measures of symmetry is well suited for implicit mode switching. Most symmetric gestures were initiated with both hands shortly after one another, but the 95th percentile of ΔT_{touch} with the symmetric *RST* technique is at 2003 ms (see P_{95} in Table 4.4). This means that mode switching based ΔT_{touch} with less then 5% false positives for naïve users, would require at least 2 seconds waiting time before touching with the second hand to initiate the asymmetric technique. This would be clearly an interruption of workflow (see [37, Chapter 5.5]). During naïve usage of the RHT_{asym} technique in our experiments, such a long ΔT_{touch} was observed only in 36% of all cases.

Table 2: Descriptive statistics of ΔT_{touch} (data in ms).

	M (SD)	Min	P_5	Mdn	P_{95}	Max
RST	577 (802)	4	17	253	2003	9491
RHT	674 (1022)	4	17	283	2639	10714
RHT_{asym}	1884 (1912)	7	51	1288	5835	14002

The recorded data on bimanual movement ratios revealed more promising results. In every frame with bimanual input, we computed the ratio of each hand's accumulated movement from the overall sum. This *move ratio* is 0.5 if the input is perfectly balanced between both hands or 1 if only one hand is moving. The mean *move ratio* tends toward 0.60 during the operation of both symmetric techniques, while it is closer to 0.85 with RHT_{asym} (Figure 4).

Due to the relatively large variance, however, the distributions of *move ratio* during the naive operation of symmetric and asymmetric techniques cannot be fully separated with an intermediate threshold. If automatic mode switching to RHT_{asym} would have been applied in our experiments based on a *move ratio*>0.75 after 30 mm accumulated input distance, this would have led to 9.3% false positives where our participants actually aimed to operate the symmetric *RST* technique. On the other hand, 68.6% of all asymmetric inputs would have been identified correctly.

Nevertheless, we concluded that mode switching based on movement symmetry is still promising since it can be expected that users adjust their input behavior if this affects the resulting virtual motion. In order to minimize false positive errors and to favor the established multitouch input behavior of the symmetric *RST* technique, we propose a *move ratio* of 0.85 for switching to the asymmetric technique. This adjustment would have slightly decreased the potential recognition rate for asymmetric modes (e.g. 61.2%), but also the potential false positive error (e.g. 4%, both after 30 mm accumulated travel). Taking this decision already after 15 mm travel makes the system more responsive, but less accurate, e.g., 5.3% false positives and 59.2% correct recognition for our naive usage data.

4.5 Frequency of Hand Postures

During the study, the number of associated fingers to each hand did not affect the system behavior in any way but we captured this data every frame for later analysis. The results indicate that our participants rarely used their hands with exactly two fingers (Table 4.5). Two-finger postures may thus be suitable to operate alternative touch-input modes. The distance between these two fingers can additonally be taken into account to further reduce the risk of involuntary activation. In almost 90% of all two-finger hand input, the distance between both fingers was shorter than 8 cm. A two-finger posture with a minimum distance of 8 cm between both fingers is comfortable to perform, but in our logs this posture occurred only in 0.13% of all recorded frames with touch input.

Table 3: Frequencies of hand postures with different numbers of associated fingers.

Num. Fingers	1	2	3	4	5
Frequency	9.5%	1.3%	3.0%	17.3%	69,0%

5 RST 3D

Based on the above-described observations and considerations, we specified a set of multitouch input gestures that supports multiscale 3D navigation with seven degrees of freedom. Figure 5 illustrates the involved input states and their behavior using the taxonomy of Martinet et al. [31]. It was our goal to remain as consistent as possible to the established RST technique for touch-based 2D user interfaces. Therefore, input from one or two hands with three or more fingers (89.3% of all cases in our logs) offers the expected 2D navigation functionality for panning along the display surface and rotation about its normal (Figure 1(a) & 1(b)). We enable scaling only with two hands, since this offers more control and accuracy on a tabletop.

Input from a single finger, which also occurs often (9.5%), maps to the same geometrical transformations as the whole hand, but in the context of our multi-user scenarios we reserve this pointing gesture for interpersonal communication. It activates a pointing ray for the indication of features below the display surface (Figure 1(d)).

A bimanual pinch gesture can also result in vertical translation instead of scaling—if it is performed asymmetrically (Figure 1(c)). In our implementation, the system switches to the asymmetric RHT mode, if one hand induced 85% of an initial bimanual motion input

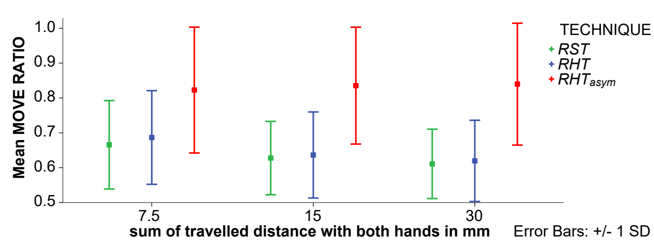


Figure 4: Means and distribution of *move ratio* for three cases of accumulated input distance and the three tested techniques.

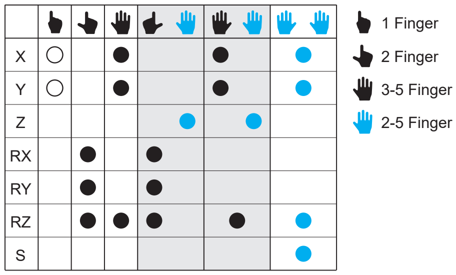


Figure 5: Illustration of multitouch input states and their 3D transformation functionalities according to the taxonomy of Martinet et al. [31]. The two-finger chord requires a minimum finger distance of 8 cm. Blue hand symbols indicate that the number of fingers does not matter. The two columns with the grey background describe asymmetric modes. Unfilled circles indicate that the input does not affect the scene in our implementation.

with an accumulated distance of 1.5 cm. Our logs from the first study indicate a chance of over 5% that naïve users without introduction to this type of mode switching, activate the asymmetric mode accidentally. Also the correct recognition rate would have been unacceptably low (less than 60%) in this case. However, we believed that users can adapt their behavior, once they have been introduced to the technique.

The most hidden mode with our gesture set is 3D rotation using a hand posture with two fingers that need to be spread apart at least 8 cm (Figure 1(e)). Our logs from the first study indicate that this gesture practically does not occur accidentally (0.13%), but we found that it can be performed quite comfortably. The associated rotation offers direct control over all three degrees of freedom.

In all input states we visualize the connection between the hand on the touch surface to the displayed scene with a virtual ray (Figure 1). In case of 3D rotation, the input motion on the touch surface is applied to rotation around the ray's intersection point as if one would tilt and turn the scene with the ray (Figure 1(e)). The spatial relation between the fingers on the display and the intersection point in the scene thus affects the rotation velocity and potentially even its direction (if the intersection point is above the user's hand). We avoid changes of rotation direction and extreme velocities by limiting the intersection point to locations that are at least 10 cm below the display surface. If the actual intersection point with the scene is found above, the technique defaults to a pivot point 10 cm below the induced touch input. More indirect mappings would allow more rapid rotations [41], but in case of unconstrained 3D rotations, this can result in confusing behavior. Turning an object upside down affects the mapping from the touch input space to object space.

6 Comparative Usability Study

The above-described multitouch 3D navigation technique was used in various student projects and a series of workshops with archeologists. Most people needed only a brief introduction to operate the system effectively. Also, during several public events, where the time for introduction was very limited, our guests found it usable. However, mode switching based on postures and quasi-postures induces cognitive load which may impair user performance in primary data exploration tasks.

We devised a user study to gain a better understanding of the cognitive load and usability of our technique and to compare it with alternative implementations for multiscale 3D navigation. Several earlier proposed multitouch 3D input methods are similar in many regards. The main difference of our approach is to consider hands as a basic input entity and to measure the symmetry of bimanual input for the distinction between vertical translation and scaling. Apart from that, other touch-based techniques also rely on mode switching to realize the required 3D transformations—often based on the number of involved fingers [18, 29, 31]. Instead of extending these techniques to support both scaling and 3D translation, we thus decided to compare our technique as a representative of such gesture-only multitouch mappings for 3D transformations against a conceptually different alternative.

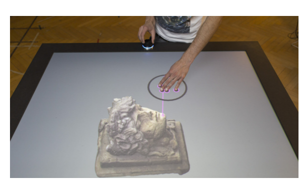
Another approach to match multitouch input to 3D transformations with 7 DOF and more is widget-based mode switching. Object-space widgets (e.g. [3, 7]), however, are not suitable for touch-based 3D navigation of content on an immersive 3D display. Without spatial congruency of the input surface and the displayed content as in case of 2D monoscopic displays, the widget controls are hardly accessible. *FI3D* by Yu et al. [50] and *Tucheo* by Hachet et al. [16], instead, build on widgets in screen space that can mediate between the 2D input surface and transformations in 3D space. A comparison with these techniques is interesting and envisaged for future work, but neither of both can be considered as being an established standard.

We thus decided to test the usability of *RST 3D* in comparison to a dedicated 3D motion controller. The Spacemouse™ is an elastic 3D rate controller that supports simultaneous 3D rotation and translation. It is the de-facto standard for 3D view navigation on desktop workstations in computer-aided design. The controller does not implicitly support scaling so we combined it with common 2D multitouch input (RST). In this sense, the study compared gesture-based mode switching with switching between dedicated devices. Control over panning and rotation about the display normal was redundantly available in the *Spacemouse* condition. Users could either perform most of the task with the Spacemouse and only operate multitouch for scaling or they could navigate primarily with multitouch and apply the Spacemouse only for vertical translation and tilt.

Multitouch interaction generally follows the paradigm of scene manipulation (scene in hand) rather than camera navigation (camera in hand). For consistency, we applied the same mapping for the Spacemouse™. The default rotation center was set to the center of the display but, since we had also touch available, we allowed an alternative rotation center to be set with the other hand on the tabletop (the intersection point of the connecting ray, see Figure 6(a)). This also constrained Spacemouse input to rotation only, while touch input allowed simultaneous 2D panning.

6.1 Task, Procedure, and Participants

We devised a multiscale 3D navigation task that required participants to rapidly reach certain destinations in the scene, acquire information, and memorize it for later recall. A highly detailed 3D scanned head sculpture was used as the scene to be explored. The search and memory task was formalized through the placement of six tiny yellow target objects on the object surface. These target objects were rectangular tubes with a hidden number inside. The study participants needed to navigate very closely, increase the object size,



(a) The *Spacemouse* condition with multitouch support for uniform scaling and the specification of a rotation center.

(b) Reaching a target destination in the *RST3D* condition. Looking inside the red box, revealed a number to be memorized.

Figure 6: The multiscale 3D navigation and information gathering task in both conditions.

and turn the scene so such that they could look at the respective number. We advised the participants to memorize as many of these numbers as possible and their location in the scene, but to focus primarily on the value and location of the highest number.

The order of targets was predefined and we asked participants to read out loud each number they identified. The experimenter confirmed each of these intermediate target acquisitions with a keystroke (spacebar) which stopped the time recording until the participants continued with navigation input. This gave the opportunity for short breaks between each subtask. The next target location was always indicated by a red highlight (Figure 6(b)). After each series of the same six targets with different numbers, the scene was reset to an overview and participants were asked if they memorized the largest number, at which target location they found it, as well as which other numbers and locations they had memorized. Each correctly memorized number and each correctly memorized location was noted by the experimenter as a bit of memorized information.

The order of conditions was balanced between two groups of participants. After an introduction to the respective technique, each participant performed two practice trials and then five trials in which we recorded the bits of information our participants could recall, as well as task completion times. The numbers in each repetition were randomized and ranged from 10 to 85. After the test with each condition, participants filled out a questionnaire with the System Usability Scale [4].

The introduction in the first condition included the multitouch interaction as well as our cut-away visualization technique for minimizing perceptual conflicts. We demonstrated to our participants the visual inconsistency when using multitouch to interact with 3D data appearing in front of the display surface and how our technique clips geometry between touching hands and the users' eyes. In order to get their honest preference, they could choose to perform the study with or without the technique activated.

We invited 16 students from our campus (9 male, 7 female; 20-40 years, M=26.29, SD=4.93). They received an allowance of 10 Euro for their participation. All but one reported using multitouch input on their smartphones on a daily basis and 12 participants had some prior experience with monoscopic multitouch tabletops. All but three participants reported prior experience with the Spacemouse with two of them claiming to be proficient users.

6.2 Results

After pilot tests with two authors of this paper, we did not expect significant differences between the conditions. We solved each task with six target locations in 25.5 s on average with *RST 3D* and 27.5 s on average in the *Spacemouse* condition and did not find the memorization task to be affected by either of the two techniques. We did expect, however, comparable performance and a confirmation of system usability in terms of SUS scores [4]. We were also keen to learn about the participants' opinion of the clipping technique.

Our results confirm that both techniques support comparable performance and similar usability (Table 6.2). The SUS scores indicate even better usability for the 3D multitouch technique, while the combination of a dedicated 3D controller and multitouch scored closer to the average value of 68 [28]. No statistically significant difference was obtained.

Table 4: Task times (TCT), memorized information (numbers and locations), and system usability scores (SUS) for both techniques.

	RST 3D	Spacemouse + RST
TCT (SD)	57.78 s (26.69)	61.89 s (SD=28.11)
Numbers (SD)	2.50 (1.18)	2.51 (1.23)
Locations (SD)	1.98 (1.22)	2.19 (1.28)
SUS (SD)	75.29 (13.11)	69.85 (19 .54)

7 Conclusion and Future Work

We presented the development of a comprehensive 3D multitouch gesture set for stereoscopic displays with multitouch support. Our gesture-only navigation technique includes control of 3D rotation, 3D translation, and uniform scaling. The design was determined by a formal analysis of the spatiotemporal motor behavior of users during symmetric and asymmetric bimanual pinch gestures which are commonly used for specifying translation in the third dimension or uniform scaling. We also suggested and implemented user-dependent clipping for situations where the users' hands intersect virtual objects which we found to reduce perceptual conflicts during interaction. The usability of our techniques was confirmed by a final usability study using our implementation on a multitouch-enabled immersive 3D tabletop display for up to three tracked users.

Our current implementation relies on diffuse infrared (IR) illumination for recognizing and tracking multiple hands with their associated fingers using the maximally extremal regions algorithm [12]. Thus the design of gestures is not limited to the number of involved fingers but it can build on more meaningful information about hands and hand postures. Future work should consider alternative approaches for such rich input sensing, e.g., heuristics based on spatiotemporal coherence and behavioral patterns.

Currently, we consider only the first two hands involved in an interaction since we do not use any heuristics to assign the hands to individual users. We rarely observed conflicts between multiple users since the particular situation at the tabletop display provides perfect awareness of who is interacting. Nevertheless, users could assist each other and e.g. provide constraints for each other's action if the gesture recognition would be user-aware.

Acknowledgments

Our research has received funding from the EC FP7 project 3D-PITOTI (ICT-2011-600545) and the German Federal Ministry of Education and Research (BMBF) under grants 031PT704X (project Big Data Analytics) and 03PSIPT5A (project *Provenance Analytics*). We thank the participants of our studies, the members of the 3D-Pitoti project, and the members and students of the Virtual Reality and Visualization Research group at Bauhaus-Universität Weimar.

References

[1] M. Agrawala, A. C. Beers, I. McDowall, B. Fröhlich, M. T. Bolas, and P. Hanrahan. The two-user responsive workbench: support for collaboration through individual views of a shared space. In *Proc. of SIGGRAPH 1997*, pp. 327–332. ACM, 1997. doi: 10.1145/258734.258875

[2] H. Benko and S. Feiner. Balloon selection: A multi-finger technique for accurate low-fatigue 3D selection. In *Proc. of IEEE Symp. 3D User Interfaces*, Mar. 2007. doi: 10.1109/3DUI.2007.340778

[3] B. Bollensdorff, U. Hahne, and M. Alexa. The effect of perspective projection in multi-touch 3d interaction. In *Proc. of GI '12*, pp. 165–172, 2012.

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

[5] G. Bruder, F. Steinicke, and W. Sturzlinger. To touch or not to touch?: Comparing 2d touch and 3d mid-air interaction on stereoscopic tabletop surfaces. In *Proc. of SUI '13*, pp. 9–16, 2013. doi: 10.1145/2491367.2491369

[6] D. Coffey, N. Malbraaten, T. Le, I. Borazjani, F. Sotiropoulos, A. Erdman, and D. Keefe. Interactive slice wim: Navigating and interrogating volume data sets using a multisurface, multitouch vr interface. *IEEE TVCG*, 18(10):1614–1626, 2012. doi: 10.1109/TVCG.2011.283

[7] A. Cohé, F. Dècle, and M. Hachet. tbox: A 3d transformation widget designed for touch-screens. In *Proc. of ACM CHI '11*, pp. 3005–3008, 2011. doi: 10.1145/1978942.1979387

[8] A. Cohé and M. Hachet. Beyond the mouse: Understanding user gestures for manipulating 3d objects from touchscreen inputs. *Computers & Graphics*, 36(8):1119 – 1131, 2012.

[9] F. Daiber, E. Falk, and A. Krüger. Balloon selection revisited: Multi-touch selection techniques for stereoscopic data. In *Proc. of ACM AVI '12*, pp. 441–444, 2012. doi: 10.1145/2254556.2254641

[10] J. B. De la Rivière, N. Dittlo, E. Orvain, C. Kervégant, M. Courtois, and T. Da Luz. ilight 3d touch: A multiview multitouch surface for 3d content visualization and viewpoint sharing. In *Proc. of ACM ITS '10*, pp. 312–312, 2010. doi: 10.1145/1936652.1936740

[11] J. Edelmann, A. Schilling, and S. Fleck. The dabr - a multitouch system for intuitive 3d scene navigation. In *Proc. of 3DTV Conference '09*, pp. 1–4, 2009. doi: 10.1109/3DTV.2009.5069671

[12] P. Ewerling, A. Kulik, and B. Froehlich. Finger and hand detection for multi-touch interfaces based on maximally stable extremal regions. In *Proc. of ITS'12*, pp. 173–182, 2012. doi: 10.1145/2396636.2396663

[13] P. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. In *Journal of Experimental Psychology 47*, pp. 381–391, 1954.

[14] C.-W. Fu, W.-B. Goh, and J. A. Ng. Multi-touch techniques for exploring large-scale 3d astrophysical simulations. In *Proc. of ACM CHI '10*, pp. 2213–2222, 2010. doi: 10.1145/1753326.1753661

[15] A. Giesler, D. Valkov, and K. Hinrichs. Void shadows: Multi-touch interaction with stereoscopic objects on the tabletop. In *Proc. of SUI '14*, pp. 104–112, 2014. doi: 10.1145/2659766.2659779

[16] M. Hachet, B. Bossavit, A. Cohé, and J.-B. de la Rivière. Toucheo: Multitouch and stereo combined in a seamless workspace. In *Proc. of UIST '11*, pp. 587–592, 2011. doi: 10.1145/2047196.2047273

[17] M. Hancock, S. Carpendale, and A. Cockburn. Shallow-depth 3d interaction: Design and evaluation of one-, two- and three-touch techniques. In *Proc. of ACM CHI '07*, pp. 1147–1156, 2007. doi: 10.1145/1240624.1240798

[18] M. Hancock, T. ten Cate, and S. Carpendale. Sticky tools: Full 6dof force-based interaction for multi-touch tables. In *Proc. of ACM ITS '09*, pp. 133–140, 2009. doi: 10.1145/1731903.1731930

[19] M. Herrlich, B. Walther-Franks, and R. Malaka. Integrated rotation and translation for 3d manipulation on multi-touch interactive surfaces. In *Proc. of SG'11*, pp. 146–154. Springer-Verlag, 2011.

[20] E. Hornecker, P. Marshall, N. S. Dalton, and Y. Rogers. Collaboration and interference: awareness with mice or touch input. In *Proc. of ACM CSCW 2008*, pp. 167–176, 2008. doi: 10.1145/1460563.1460589

[21] S. Houde. Iterative design of an interface for easy 3-d direct manipulation. In *Proc. of ACM CHI '92*, pp. 135–142, 1992. doi: 10.1145/142750.142772

[22] H. Hua, L. Brown, and C. Gao. Scape: supporting stereoscopic collaboration in augmented and projective environments. *Computer Graphics and Applications, IEEE*, 24(1):66–75, Jan-Feb 2004. doi: 10.1109/MCG.2004.1255811

[23] P. Isenberg, D. Fisher, S. A. Paul, M. R. Morris, K. Inkpen, and M. Czerwinski. Co-located collaborative visual analytics around a tabletop display. *IEEE TVCG*, 18(5):689–702, May 2012. doi: 10.1109/TVCG.2011.287

[24] T. Isenberg. Interactive exploration of three-dimensional scientific visualizations on large display surfaces. In C. Anslow, P. Campos, and J. Jorge, eds., *Collaboration Meets Interactive Spaces*, pp. 97–123. Springer Int. Publishing, 2016. doi: 10.1007/978-3-319-45853-3_6

[25] T. Isenberg and M. Hancock. Gestures vs. Postures: 'Gestural' Touch Interaction in 3D Environments. In K. Anderson, L. Arhippainen, H. Benko, J.-B. de la Rivière, J. Häkkilä, A. Krüger, D. F. Keefe, M. Pakanen, and F. Steinicke, eds., *Proc. of 3DCHI Workshop*, pp. 53–61. published online, United States, May 2012.

[26] K. Kin, T. Miller, B. Bollensdorff, T. DeRose, B. Hartmann, and M. Agrawala. Eden: A professional multitouch tool for constructing virtual organic environments. In *Proc. of ACM CHI '11*, pp. 1343–1352, 2011. doi: 10.1145/1978942.1979141

[27] A. Kulik, A. Kunert, S. Beck, R. Reichel, R. Blach, A. Zink, and B. Fröhlich. C1x6: a stereoscopic six-user display for co-located collaboration in shared virtual environments. *ACM Trans. Graph.*, 30(6):188:1–188:12, Dec. 2011. doi: 10.1145/2070781.2024222

[28] J. R. Lewis and J. Sauro. The factor structure of the system usability scale. In *Proc. of HCI International 2009*, HCD 09, pp. 94–103. Springer-Verlag, 2009. doi: 10.1007/978-3-642-02806-9_12

[29] J. Liu, O. K.-C. Au, H. Fu, and C.-L. Tai. Two-finger gestures for 6dof manipulation of 3d objects. *Computer Graphics Forum*, 31(7):2047–2055, 2012. doi: 10.1111/j.1467-8659.2012.03197.x

[30] D. Marchal, C. Moerman, G. Casiez, and N. Roussel. Designing intuitive multi-touch 3d navigation techniques. In *Proc. of INTERACT 2013*, vol. 8117 of *Lecture Notes in Computer Science*, pp. 19–36. Springer Berlin Heidelberg, 2013. doi: 10.1007/978-3-642-40483-2_2

[31] A. Martinet, G. Casiez, and L. Grisoni. Integrality and separability of multitouch interaction techniques in 3d manipulation tasks. *IEEE TVCG*, 18(3):369–380, March 2012. doi: 10.1109/TVCG.2011.129

[32] F. Marton, M. Agus, E. Gobbetti, G. Pintore, and M. B. Rodriguez. Natural exploration of 3d massive models on large-scale light field displays using the {FOX} proximal navigation technique. *Computers & Graphics*, 36(8):893–903, 2012. doi: 10.1016/j.cag.2012.06.005

[33] M. R. Masliah and P. Milgram. Measuring the allocation of control in a 6 degree-of-freedom docking experiment. In *Proc. of ACM CHI '00*, pp. 25–32, 2000. doi: 10.1145/332040.332403

[34] C. Moerman, D. Marchal, and L. Grisoni. Drag'n go: Simple and fast navigation in virtual environment. In *Proc. of IEEE 3DUI '12*, pp. 15–18, March 2012. doi: 10.1109/3DUI.2012.6184178

[35] M. Möllers, P. Zimmer, and J. Borchers. Direct manipulation and the third dimension: Co-planar dragging on 3d displays. In *Proc. of ACM ITS '12*, pp. 11–20, 2012. doi: 10.1145/2396636.2396639

[36] M. A. Nacenta, P. Baudisch, H. Benko, and A. Wilson. Separability of spatial manipulations in multi-touch interfaces. In *Proc. of GI '09*, pp. 175–182, 2009.

[37] J. Nielsen. *Usability engineering*. Elsevier, 1994.

[38] P. Paczkowski, J. Dorsey, H. Rushmeier, and M. H. Kim. Paper3d: Bringing casual 3d modeling to a multi-touch interface. In *Proc. of ACM UIST '14*, pp. 23–32, 2014. doi: 10.1145/2642918.2647416

[39] J. L. Reisman, P. L. Davidson, and J. Y. Han. A screen-space formulation for 2d and 3d direct manipulation. In *Proc. of ACM UIST '09*, UIST '09, pp. 69–78, 2009. doi: 10.1145/1622176.1622190

[40] Y. Rogers and S. Lindley. Collaborating around vertical and horizontal large interactive displays: which way is best? *Interacting w. Computers*, 16(6):1133–1152, 2004. doi: 10.1016/j.intcom.2004.07.008

[41] E. Rousset, F. Bérard, and M. Ortega. Two-finger 3d rotations for novice users: Surjective and integral interactions. In *Proc. of AVI '14*, pp. 217–224. ACM, 2014. doi: 10.1145/2598153.2598183

[42] S. Schneegans, F. Lauer, A. Bernstein, A. Schollmeyer, and B. Froehlich. guacamole - an extensible scene graph and rendering framework based on deferred shading. In *7th IEEE SEARIS '14*, pp. 35–42, 2014.

[43] K. Shoemake. Arcball: a user interface for specifying three-dimensional orientation using a mouse. In *GI '92*, pp. 151–156, 1992.

[44] A. L. Simeone. Indirect touch manipulation for interaction with stereoscopic displays. In *Proc. of IEEE Symp. 3D User Interfaces (3DUI)*, pp. 13–22, Mar. 2016. doi: 10.1109/3DUI.2016.7460025

[45] S. Strothoff, D. Valkov, and K. Hinrichs. Triangle cursor: Interactions with objects above the tabletop. In *Proc. of ACM ITS '11*, pp. 111–119, 2011. doi: 10.1145/2076354.2076377

[46] R. Teather and W. Stuerzlinger. Pointing at 3d targets in a stereo head-tracked virtual environment. In *Proc. of 3DUI '11*, pp. 87–94, March 2011. doi: 10.1109/3DUI.2011.5759222

[47] D. Valkov, A. Giesler, and K. Hinrichs. Evaluation of depth perception for touch interaction with stereoscopic rendered objects. In *Proc. of ACM ITS '12*, pp. 21–30, 2012. doi: 10.1145/2396636.2396640

[48] D. Valkov, F. Steinicke, G. Bruder, and K. Hinrichs. 2d touching of 3d stereoscopic objects. In *Proc. of ACM CHI '11*, pp. 1353–1362, 2011. doi: 10.1145/1978942.1979142

[49] M. Veit, A. Capobianco, and D. Bechmann. Dynamic decomposition and integration of degrees of freedom for 3-d positioning. In *Proc. of VRST'10*, pp. 131–134, 2010. doi: 10.1145/1889863.1889891

[50] L. Yu, P. Svetachov, P. Isenberg, M. H. Everts, and T. Isenberg. Fi3d: Direct-touch interaction for the exploration of 3d scientific visualization spaces. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1613–1622, Nov 2010. doi: 10.1109/TVCG.2010.157

[51] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, Oct 1994.